



Numerical Multilevel Upscaling for Incompressible Flow in Reservoir Simulation: An Element-based Algebraic Multigrid (AMGe) Approach

Christensen, Max la Cour; Villa, Umberto; Engsig-Karup, Allan Peter; Vassilevski, Panayot S.

Published in:
SIAM Journal on Scientific Computing

Link to article, DOI:
[10.1137/140988991](https://doi.org/10.1137/140988991)

Publication date:
2017

Document Version
Publisher's PDF, also known as Version of record

[Link back to DTU Orbit](#)

Citation (APA):
Christensen, M. L. C., Villa, U., Engsig-Karup, A. P., & Vassilevski, P. S. (2017). Numerical Multilevel Upscaling for Incompressible Flow in Reservoir Simulation: An Element-based Algebraic Multigrid (AMGe) Approach. *SIAM Journal on Scientific Computing*, 39(1), B102-37. <https://doi.org/10.1137/140988991>

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

NUMERICAL MULTILEVEL UPSCALING FOR INCOMPRESSIBLE FLOW IN RESERVOIR SIMULATION: AN ELEMENT-BASED ALGEBRAIC MULTIGRID (AMGe) APPROACH*

MAX LA COUR CHRISTENSEN[†], UMBERTO VILLA[‡], ALLAN P. ENGSIG-KARUP[§], AND
PANAYOT S. VASSILEVSKI[¶]

Abstract. We study the application of a finite element numerical upscaling technique to the incompressible two-phase porous media total velocity formulation. Specifically, an element-agglomeration-based algebraic multigrid (AMGe) technique with improved approximation properties [I. Lashuk and P. Vassilevski, *Numer. Linear Algebra Appl.*, 19 (2012), pp. 414–426] is used, for the first time, to generate upscaled and accurate coarse systems for the reservoir simulation equations. The upscaling technique is applied to both the mixed system for velocity and pressure and to the hyperbolic transport equations, providing fully upscaled systems. By introducing additional degrees of freedom associated with nonplanar interfaces between agglomerates, the coarse velocity space has guaranteed approximation properties. The employed AMGe technique provides coarse spaces with desirable local mass conservation and stability properties analogous to the original pair of Raviart–Thomas and piecewise discontinuous polynomial spaces, resulting in strong mass conservation for the upscaled systems. Due to the guaranteed approximation properties and the generic nature of the AMGe method, recursive multilevel upscaling is automatically obtained. Furthermore, this technique works for both structured and unstructured meshes. Multiscale mixed finite elements exhibit accuracy for general unstructured meshes but do not in general lead to nested hierarchy of spaces. Multiscale multilevel mimetic finite differences generate nested spaces but lack the adaptivity of the flux representation on coarser levels that the proposed AMGe approach offers. Thus, the proposed approach can be seen as a rigorous bridge that merges the best properties of these two existing methods. The accuracy and stability of the studied multilevel AMGe upscaling technique is demonstrated on two challenging test cases.

Key words. element-based algebraic multigrid, numerical upscaling, multilevel upscaling, reservoir simulation, mixed finite element method, discontinuous Galerkin finite element method, porous media flow, subsurface flow

AMS subject classifications. 65M60, 65M55, 76S05

DOI. 10.1137/140988991

1. Introduction. Upscaling of geological properties is an essential practice in reservoir simulation, since the spatial resolution of the geological model often is too

*Submitted to the journal's Computational Methods in Science and Engineering section September 22, 2014; accepted for publication (in revised form) August 18, 2016; published electronically February 23, 2017. This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under contract DE-AC52-07NA27344 supported in part by the U.S. Department of Energy, Office of Science, Office of Advanced Scientific Computing Research, Applied Mathematics program. Accordingly, the U.S. Government retains a nonexclusive, royalty-free license to publish or reproduce the published form of this contribution, or allow others to do so, for U.S. Government purposes.

<http://www.siam.org/journals/sisc/39-1/98899.html>

[†]Department of Applied Mathematics and Computer Science, Technical University of Denmark, 2800 Kongens Lyngby, Denmark, and Center for Energy Resources Engineering (CERE), Technical University of Denmark, Lloyd's Register Consulting, 2900 Hellerup, Denmark (max@maxlacour.com).

[‡]Center for Applied Scientific Computing, Lawrence Livermore National Laboratory, Livermore, CA 94551. Current address: Institute for Computational Engineering and Sciences, The University of Texas at Austin, Austin, TX 78712 (uvilla@ices.utexas.edu).

[§]Department of Applied Mathematics and Computer Science, Technical University of Denmark, 2800 Kongens Lyngby, Denmark (apek@dtu.dk).

[¶]Center for Applied Scientific Computing, Lawrence Livermore National Laboratory, Livermore, CA 94551 (vassilevski1@llnl.gov).

high for reservoir simulators to execute in practical times. The traditional approach employed today resorts to computing effective properties of the subsurface (e.g., permeability and porosity) by various homogenization, averaging, or upscaling techniques. The so-called flow-based upscaling methods are among the most used. They are typically based on solving a simple steady-state elliptic differential equation. Given the solution of this equation, effective coarse permeabilities can be computed. Some flow-based upscaling methods provide full tensor coarse permeabilities, but in practice mostly diagonal tensor permeabilities are used. These effective coarse properties are then perceived as the “true” model from this point on. However, the use of upscaling in the workflow introduces a black box step, where the relation/difference between the solution of the upscaled model and the solution of the fine grid model (geological resolution) is difficult (or impossible) to determine. A significant body of research has gone into improving this workflow by introducing new upscaling methods that establish a connection between the fine- and coarse-scale models. In the petroleum engineering community, these methods are typically referred to as multiscale methods. The success of these methods can be partially understood by their relationship to homogenization, which performs a formal averaging of scale-separated processes (i.e., averaging of the underlying mathematical operators) as opposed to simply averaging the parameters of the model. Although this concept of averaging (or coarsening) an operator is generally understated, it is a cornerstone of multiscale methods and helps motivate the new element-based algebraic multigrid (AMGe) approach, as well as other multilevel operator-based approaches.

The methods described in this paper are strongly related to the multiscale mixed finite element method (MsMFEM). The MsMFEM stems from early work described in [27] and [15], where specific finite element basis functions were used to construct a tool for multiscale solution of elliptic partial differential equations (PDEs) in both primal and mixed forms. Since then, much research has been carried out on this topic to improve the approximation properties and extend the range of physical phenomena described by the models [5, 7, 6, 1, 35]. Among other things, the method was extended to achieve locally mass conservative velocity fields on the subgrid scale, which enabled a combination of the MsMFEM and streamline simulations [3]. Other work focuses on updating the multiscale basis functions for time-dependent problems [34] or to capture specific features of the flow [2]. Multiscale methods have also attracted attention for locally conservative mimetic finite difference methods [4] and for finite volume methods [29, 23]. Adaptive strategies for multiscale techniques have also been proposed [23, 40].

Multiscale methods have been extended for mimetic finite differences to work in a multilevel way for two-phase flow problems [39, 40, 41]. Multiscale multilevel mimetic methods, namely M^3 , have several similarities to our approach and a few differences. Important similarities include the ability to handle aggregates with nonplanar faces, achieve local mass conservation on all levels, and provide support for well models. The AMGe approach possesses all these properties with the additional flexibility to assign a variable number of degrees of freedom per agglomerated face (interface between two agglomerated elements) that is automatically determined by the desired accuracy and by the topology of the agglomerated face by means of SVD.

Finally, the multilevel upscaling technique introduced in [42] leverages the components from a multigrid algorithm, using algebraically constructed coarse spaces and variational Galerkin coarsening. Our AMGe approach exhibits similar features with the additional caveat that we ensure that the coarse spaces maintain guaranteed order of approximation at all levels.

In parallel to the above cited multiscale methods, in the algebraic multigrid (AMG) community, the construction of coarse problems was an essential component in developing efficient multilevel solvers for the fine-grid (fine-scale) problems of interest, especially in the unstructured mesh setting. It was recognized for quite some time that an efficient two-grid (TG) solver requires as a necessary condition a coarse space that admits certain weak approximation properties. For a rigorous proof of this fact we refer the reader to [25]. This fact can be viewed as a cornerstone motivating point in using AMG-constructed finite element coarse spaces as discretization spaces, i.e., as a tool for numerical upscaling. We refer the interested reader to the overview in [56] for more details. For some early work on using operator-dependent (AMG) coarse spaces for numerical homogenization, we refer the reader to [33]. Among many AMG-type coarse spaces that can be constructed, we are interested in ones that can handle general classes of finite element spaces and hence be applicable to broad classes of PDEs, namely, the spaces that form a de Rham complex (i.e., the sequence of H^1 -conforming, $H(\text{curl})$ -conforming, $H(\text{div})$ -conforming, and L^2 -conforming spaces) with applications to elliptic PDEs, Maxwell equations, Darcy flow equations, etc. The work [49], although specifically motivated to construct coarse de Rham complexes for use in multigrid solvers, provided the basis for extensions finalized in [38], to build coarse spaces with guaranteed approximation properties, giving rise to an efficient upscaling tool. Since our construction of coarse spaces applies to the entire de Rham sequence, the developed technique can also be used for other applications, such as the mixed formulation of the Brinkman problem [53, 57].

Often, multiscale methods for reservoir simulation solve for the pressure (and velocity) on a coarse scale and keep the saturation equations on the fine grid. With this approach, solving the saturation equations quickly becomes the dominant bottleneck. Methods have been developed to also upscale the saturation equation [59, 24]. Our approach enables upscaling of not only the mixed system for velocity and pressure but also the transport equations for the saturations using the same framework. In the present paper, the coarse space used for the pressure is reused for the saturation equations. For problems involving quantities of interest that do not require a fine-scale solution of the saturation equations, such fully upscaled models can greatly accelerate standard methods in uncertainty quantification (e.g., using multilevel Monte Carlo [20, 21]) and optimization (e.g., MG/OPT [11]) due to the ability to simulate with good accuracy at different levels of resolution with a reduction in computational cost equivalent to the reduction in the degrees of freedom for the upscaled models. AMGe has been developed (in LLNL) since its introduction (see [30, 54, 13, 14, 37]) as a general multilevel coarsening framework with a wide range of applications. In addition to numerical upscaling, it is also designed to create efficient and optimal solvers that can be adapted throughout the simulation [31]. Furthermore, the AMGe upscaling technique targets general unstructured meshes as well as higher-order elements. As such, it is distinctly different (more general) than the above referred multiscale and mimetic methods.

For large-scale models and for applications such as uncertainty quantification and optimization, TG upscaling is insufficient. If the fine grid problem contains a large number of elements and more aggressive coarsening is needed to keep the upscaled problem small, constructing coarse spaces becomes increasingly expensive due to the growing sizes of the local flow problems. Furthermore, in uncertainty quantification (e.g., multilevel Monte Carlo methods) and optimization, it is beneficial to employ a hierarchy of coarser systems to accelerate the convergence. For this reason, multilevel upscaling is a natural choice, since it enables the size of the local flow problems, needed

to construct the coarse spaces, to remain small (even fixed) and it provides a hierarchy of coarser discretizations. The AMGe technique employed in the present work allows a completely recursive upscaling, where a hierarchy of agglomerated meshes is used, and due to the fact that we upscale the full system of equations, simulations can be carried out for a range of coarser representations, where the computational cost of a simulation is reduced in a similar fashion as the number of degrees of freedom and the nonzeros are reduced. The multilevel upscaling makes it possible to reuse the hierarchy of coarser spaces for different purposes. It can be used for both numerical upscaling, linear solvers, and even nonlinear solvers [12].

The contribution of the present paper is in the application of one version of AMGe with guaranteed approximation properties [36, 38, 49] to the incompressible two-phase flow equations for reservoir simulation. Moreover, the framework is completely recursive, allowing for multilevel upscaling with guaranteed approximation properties. The paper demonstrates multilevel upscaling for two challenging test cases. In addition to upscaling the mixed system for pressure and velocity, the saturation equations are also upscaled with the same coarse spaces used for the pressure. To our knowledge there is no other method that supports mixed finite element formulations on general unstructured grids; it allows for multilevel nested hierarchies as well as allowing for great flexibility in the construction of the coarse spaces—with two possible strategies to locally enrich the coarse spaces by either using finer agglomerates or adding additional degrees of freedom for each agglomerate.

The remainder of the paper is structured as follows. In section 2, we introduce the governing equations and present the weak formulation of the problem. In section 3, the system of equations is discretized in space and time. In section 4, the improved AMGe is introduced. Finally, in section 5, numerical results are presented for two challenging test cases. The paper concludes with a summary and perspectives in sections 6 and 7.

2. Governing equations. In this section, we briefly introduce a total velocity formulation of a simplified model for multiphase flow in porous media. We refer the reader to [17] for a rigorous derivation of this formulation from the physics principles of mass and momentum conservation. The unknowns of the formulation are the total velocity \mathbf{u} , the pressure p , and the set S of the saturations S_α for each phase α . For example, we can have $\alpha = o, w, g$, and hence $S = \{S_o, S_w, S_g\}$, where o stands for oil, w stands for water, and g stands for gas. Neglecting the effects of capillary pressure and assuming incompressible rock and fluids, the system of equations is given by

$$(2.1) \quad \mathbf{K}^{-1} \lambda^{-1}(S) \mathbf{u} + \nabla p = \left(\sum_{\alpha} \rho_{\alpha} f_{\alpha}(S) \right) g \nabla z,$$

$$(2.2) \quad \nabla \cdot \mathbf{u} = q(S, p),$$

$$(2.3) \quad \phi \frac{\partial S_{\alpha}}{\partial t} + \nabla \cdot \mathbf{u}_{\alpha}(S, \mathbf{u}) = \frac{q_{\alpha}(S, p)}{\rho_{\alpha}},$$

where $\nabla \cdot$ is the divergence operator, \mathbf{K} is the absolute permeability tensor, ρ^{α} is the mass density, ϕ is the porosity, g is the gravitational acceleration, and ∇z stands for the coordinate vector in the z -direction. In this work, the capillary pressure is ignored for simplicity. Multiscale methods taking into consideration capillary pressure have been developed [48]. Since the terms and variables introduced in a formulation with capillary pressure are covered by the spaces that form a de Rham complex, we expect AMGe to be able to handle this as well.

The total velocity \mathbf{u} is the sum of the phase velocities \mathbf{u}_α :

$$(2.4) \quad \mathbf{u} = \sum_{\alpha} \mathbf{u}_{\alpha}.$$

Similarly, the total source term $q(S, p)$ is the sum of the phase source terms q_α :

$$(2.5) \quad q = \sum_{\alpha} \frac{q_{\alpha}(S, p)}{\rho_{\alpha}}.$$

The total mobility λ is given by

$$(2.6) \quad \lambda = \sum_{\alpha} \lambda_{\alpha} = \sum_{\alpha} \frac{k_{r,\alpha}(S_{\alpha})}{\mu_{\alpha}}.$$

Here, for simplicity, we assume the “straight relative permeabilities” model, and we let the relative permeability $k_{r,\alpha}$ be equal to S_{α} . The fractional flow function f_{α} is given by

$$(2.7) \quad f_{\alpha} = \frac{\lambda_{\alpha}(S_{\alpha})}{\lambda(S)}.$$

Finally, the phase velocity \mathbf{u}_{α} is related to the total velocity \mathbf{u} by

$$(2.8) \quad \mathbf{u}_{\alpha} = (\mathbf{u} + \mathbf{G}_{\alpha}) f_{\alpha},$$

where

$$(2.9) \quad \mathbf{G}_{\alpha} = \sum_{\substack{\beta=o,w,g \\ \beta \neq \alpha}} \lambda_{\beta}(\rho_{\alpha} - \rho_{\beta}) \mathbf{K} g \nabla z$$

allows for countercurrent flow due to gravity.

The total velocity formulation can be advantageous, as it allows for a less coupled system compared to other formulations [17, p. 25]. Unlike the individual phase mobility, the total mobility is positive definite, meaning its inverse always exists. Furthermore, the total velocity is smoother than a phase velocity. For these reasons, it is a good choice as a primary variable.

2.1. Wells. To model injection and production wells we use the well-understood Peaceman equations [50], which impose a linear relationship between the bottom hole pressure p_{bh} and the injection/production rates q_{α} . These formulas hold for vertical wells in anisotropic rock media with multiphase fluids and gravity and read as

$$(2.10) \quad q_{\alpha} = \mathcal{WI} \lambda_{\alpha}(p_{\text{bh}} - p - \rho_{\alpha} g(z_{\text{bh}} - z)),$$

where z_{bh} is the well datum level depth and z is the depth. The well index \mathcal{WI} is a lumped parameter that takes into account the geometry of the perforation, the geometry of the cell, and the anisotropy of the permeability tensor.

In our simulator, we use the bottom hole pressure, p_{bh} , to control the production wells, whereas we prescribe a rate for injection wells. In practice, this means

$$(2.11) \quad q_{\alpha} = \begin{cases} \text{rate} & \text{if injector,} \\ \mathcal{WI} \lambda_{\alpha}(p_{\text{bh}} - p - \rho_{\alpha} g(z_{\text{bh}} - z)) & \text{if producer,} \end{cases}$$

where *rate* is a user-given input. For producer wells, this means we have a dependence on pressure in the reservoir, which should be accounted for in the numerical schemes.

For completeness, we also provide the well-known formula to compute the well index \mathcal{WI} derived in [50] under some simplifying assumptions on the structure of the permeability tensor and on the cell geometry. For a more thorough description of well models for finite element methods, see [17, p. 450], while for more complex well models we refer the reader to [22].

Assuming the permeability tensor has a diagonal structure, $\mathbf{K} = \text{diag}(k_x, k_y, k_z)$, and that the element containing the well is a cuboid fully perforated in the vertical direction, the Peaceman formula for the well index reads as

$$(2.12) \quad \mathcal{WI} = \frac{2\pi h_3 \sqrt{k_x k_y}}{\ln(r_e/r_w) + s_k}.$$

Here, s_k is the skin factor used to model formation damage from drilling, r_w is the well radius, and the equivalent radius r_e is

$$(2.13) \quad r_e = \frac{0.14 \left(h_1^2 \sqrt{k_y/k_x} + h_2^2 \sqrt{k_x/k_y} \right)}{0.5 \left((k_y/k_x)^{1/4} + (k_x/k_y)^{1/4} \right)},$$

with h_1 , h_2 , and h_3 being the mesh sizes in the x -, y -, and z -directions.

2.2. Weak formulation. We now introduce some notation used throughout the paper. Let Ω be a bounded connected domain in \mathbb{R}^d with a regular (Lipschitz continuous) boundary $\partial\Omega$, which has a well-defined unit outward normal vector $\mathbf{n} \in \mathbb{R}^d$. For the cases considered in this paper, $d = 3$.

For the vectorial functions $\mathbf{u}, \mathbf{v} \in \mathbf{L}^2(\Omega) = [L^2(\Omega)]^d$ and scalar functions $p, w \in L^2(\Omega)$, we define the inner products $(\mathbf{u}, \mathbf{v}) = \int_{\Omega} \mathbf{u} \cdot \mathbf{v} \, d\Omega$ and $(p, w) = \int_{\Omega} p w \, d\Omega$. Finally, we introduce the functional space $H(\text{div}; \Omega)$ defined as

$$H(\text{div}; \Omega) := \{ \mathbf{u} \in \mathbf{L}^2(\Omega) \mid \text{div } \mathbf{u} \in L^2(\Omega) \}.$$

For simplicity, we assume a no-flow boundary condition for the total velocity \mathbf{u} , and we impose

$$(2.14) \quad \mathbf{u} \cdot \mathbf{n} = 0.$$

This boundary condition is the most widely used in reservoir simulation; however, different boundary conditions can be easily accommodated, including Dirichlet boundary conditions for the pressure, or more sophisticated conditions obtained by coupling the equation of reservoir with an aquifer model.

We finally introduce the functional spaces \mathcal{R} and \mathcal{W} , which are defined as

$$\begin{aligned} \mathcal{R} &\equiv \{ \mathbf{u} \in H(\text{div}; \Omega) \mid \mathbf{u} \cdot \mathbf{n} = 0 \text{ on } \partial\Omega \}, \\ \mathcal{W} &\equiv L^2(\Omega). \end{aligned}$$

To derive the weak formulation for the mixed system in (2.1) and (2.2) we multiply (2.1) and (2.2) with the test functions $\mathbf{v} \in \mathcal{R}$ and $w \in \mathcal{W}$ and integrate over the domain Ω . After integration by parts of the nonconforming terms, applying the boundary condition $\mathbf{u} \cdot \mathbf{n} = \mathbf{v} \cdot \mathbf{n} = 0$, we obtain the following variational problem.

PROBLEM 1 Find $(\mathbf{u}, p) \in \mathcal{R} \times \mathcal{W}$ such that

$$\begin{cases} \left(\mathbf{K}^{-1} \lambda^{-1}(S) \mathbf{u}, \mathbf{v} \right) - \left(p, \nabla \cdot \mathbf{v} \right) = \left(g \nabla z \sum_{\alpha} f_{\alpha}(S) \rho_{\alpha}, \mathbf{v} \right) & \forall \mathbf{v} \in \mathcal{R}, \\ \left(\nabla \cdot \mathbf{u}, w \right) - \left(q(p, S), w \right) = 0 & \forall w \in \mathcal{W}. \end{cases}$$

In Problem 1, the total mobility λ and the fractional flow function f_{α} depend on the phase saturations S , while, for the producer wells, q has a dependence on pressure p and the saturations S as given by (2.11). Problem 1 is well posed: the pair $(\mathcal{R}, \mathcal{W})$ is *inf-sup compatible*, and $\frac{\partial q}{\partial p} \leq 0$ for all pressure p and saturations S .

In a simpler way, the weak formulation for the conservation law is derived by multiplying (2.3) with the test function $w \in \mathcal{W}$ and integrating over the domain Ω .

PROBLEM 2 Find $S_{\alpha} \in \mathcal{W}$ such that

$$\left(\phi \frac{\partial S_{\alpha}}{\partial t}, w \right) + \left(\nabla \cdot \mathbf{u}_{\alpha}(S, \mathbf{u}), w \right) = \left(\frac{q_{\alpha}(p, S_{\alpha})}{\rho_{\alpha}}, w \right) \quad \forall w \in \mathcal{W}.$$

In Problem 2, the phase velocities $\mathbf{u}_{\alpha=o,g,w}$ depend on the total velocity \mathbf{u} and the saturations S as given by (2.8).

3. Discretization. For spatial discretization, the mixed finite element method (mixed FEM) is used to discretize (2.1) and (2.2), whereas the discontinuous Galerkin method is used to discretize the conservation law in (2.3). For temporal integration the improved IMPES method [16] is used to decouple the computation of the total velocity and pressure from the computation of the saturations. For simplicity, the forward Euler method is chosen as a time integrator to advance the saturation equations, since the focus of this paper is on the numerical upscaling and to demonstrate scalability in the spatial discretization. While this choice may affect the total numerical efficiency due to the global CFL condition imposed by the explicit time stepping, AMGe techniques can readily be used together with higher-order and more accurate time discretization schemes.

We stress the fact that all variables and parameters in this section are on the fine grid level, and no upscaling is introduced until section 4.

3.1. Spatial discretization of the mixed system. The mixed FEM is used to discretize (2.1) and (2.2). In particular, we let $\mathcal{R}_h \subset \mathcal{R}$ be the (lowest-order) Raviart–Thomas finite element space consisting of vector functions with a continuous normal component across the interfaces between the elements and let $\mathcal{W}_h \subset \mathcal{W}$ be the space of piecewise discontinuous polynomial (constant) scalar functions. It is well known that this choice of finite element spaces satisfies the Ladyzhenskaya–Babuška–Brezzi conditions and therefore allows for a stable discretization of Problem 1. The Galerkin formulation of the problem reads as follows.

PROBLEM 3 Find $(\mathbf{u}_h, p_h) \in \mathcal{R}_h \times \mathcal{W}_h$ such that

$$\begin{cases} \left(\mathbf{K}_h^{-1} \lambda_h^{-1} \mathbf{u}_h, \mathbf{v}_h \right) - \left(p_h, \nabla \cdot \mathbf{v}_h \right) = \left(g \nabla z_h \sum_{\alpha} f_{\alpha,h} \rho_{\alpha,h}, \mathbf{v}_h \right) & \forall \mathbf{v}_h \in \mathcal{R}_h, \\ \left(\nabla \cdot \mathbf{u}_h, w_h \right) - \left(q(p_h), w_h \right) = 0 & \forall w_h \in \mathcal{W}_h. \end{cases}$$

Here, to simplify the notation, we have omitted the dependence of λ_h , $f_{\alpha,h}$, and $q_{\alpha,h}$ on the saturations S_h .

3.1.1. Matrix form. Let us denote with $\{\phi^j\}_{j=1, \dots, \dim(\mathcal{R}_h)}$ a basis for the space \mathcal{R}_h and with $\{\psi^j\}_{j=1, \dots, \dim(\mathcal{W}_h)}$ a basis for the space \mathcal{W}_h . With this notation, the finite element solution (\mathbf{u}_h, p_h) can be written as a linear combination of the basis functions (ϕ^j, ψ^j) . More specifically, letting $\mathbf{U} \in \mathbb{R}^{\dim(\mathcal{R}_h)}$ and $P \in \mathbb{R}^{\dim(\mathcal{W}_h)}$ denote the vectors collecting the finite element degrees of freedom \mathbf{u}_h^i , $i = 1, \dots, \dim(\mathcal{R}_h)$, and p_h^i , $i = 1, \dots, \dim(\mathcal{W}_h)$, we write

$$(3.1) \quad \mathbf{u}_h = \sum_{j=1}^{\dim(\mathcal{R}_h)} \mathbf{u}_h^j \phi^j, \quad p_h = \sum_{j=1}^{\dim(\mathcal{W}_h)} p_h^j \psi^j.$$

We introduce the finite element matrices M , B whose entries are given by

$$\begin{aligned} M_{ij} &= (\mathbf{K}_h^{-1} \lambda_h^{-1} \phi^j, \phi^i), & i, j &= 1, \dots, \dim(\mathcal{R}_h), \\ B_{ij} &= (\nabla \cdot \phi^j, \psi^i), & i &= 1, \dots, \dim(\mathcal{W}_h), j = 1, \dots, \dim(\mathcal{R}_h). \end{aligned}$$

Finally, we introduce the matrix C that represents the (linear) dependence of the well production rates on the pressure. C is a diagonal semipositive definite matrix with nonzero entries only in the rows corresponding to elements containing a production well. More specifically, we have

$$C_{ij} = (\beta \psi^j, \psi^i), \quad i, j = 1, \dots, \dim(\mathcal{W}_h),$$

where $\beta = \sum_{\alpha} \mathcal{W} I_{\alpha} \chi_{\text{prod}}$ and χ_{prod} is an indicator function with support on the elements that contain a production well.

Problem 3 leads to the solution of the sparse linear system

$$(3.2) \quad \mathbf{A} \mathbf{X} = \mathbf{B},$$

where the block matrix \mathbf{A} and block vectors \mathbf{X} and \mathbf{B} read as

$$(3.3) \quad \mathbf{A} = \begin{bmatrix} M & B^T \\ B & -C \end{bmatrix}, \quad \mathbf{X} = \begin{bmatrix} \mathbf{U} \\ P \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} \mathbf{F}_u \\ \mathbf{F}_p \end{bmatrix}.$$

The linear system (3.2) is an indefinite saddle point problem [10] whose solvability is guaranteed by the fact that M and $C + BB^T$ are symmetric positive definite matrices. If no production wells were present, then $C = 0$ and the pressure would be defined up to a constant.

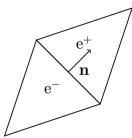


FIG. 1. Convention used in jump notation.

3.2. Spatial discretization of the saturation equations. The conservation law in (2.3) is discretized using the discontinuous Galerkin method. Integration by parts is applied to the flux term, resulting in

$$(3.4) \quad \begin{aligned} (\nabla \cdot \mathbf{u}_{\alpha,h}(S_h, \mathbf{u}_h), w_h) &= \sum_{e \in \mathcal{T}_h} \left(- \int_e \mathbf{u}_{\alpha,h}(S_h, \mathbf{u}_h) \cdot \nabla w_h \, d\Omega \right) \\ &\quad + \sum_{e \in \mathcal{T}_h} \sum_{f \in \partial e} \int_f (\mathbf{u}_{\alpha,h}(S_h, \mathbf{u}_h) \cdot \mathbf{n}_e)^* w_h \, dS, \end{aligned}$$

where e is an element in the mesh \mathcal{T}_h and f is a face in the set \mathcal{F}_h of all the faces (both boundary and internal faces) in the mesh. The star denotes the numerical flux $(\mathbf{u}_{\alpha,h}(S_h, \mathbf{u}_h) \cdot \mathbf{n}_e)^*$, which will be specified in section 3.2.1. By introducing the jump notation $[w_h] = w_h^- - w_h^+$, with the convention depicted in Figure 1, we can rewrite (3.4) as

$$(3.5) \quad \begin{aligned} (\nabla \cdot \mathbf{u}_{\alpha,h}(S_h, \mathbf{u}_h), w_h) &= - \sum_{e \in \mathcal{T}_h} \int_e \mathbf{u}_{\alpha,h}(S_h, \mathbf{u}_h) \cdot \nabla w_h \, d\Omega \\ &\quad + \sum_{f \in \mathcal{F}_h} \int_f (\mathbf{u}_{\alpha,h}(S_h, \mathbf{u}_h) \cdot \mathbf{n}_e)^* [w_h] \, dS. \end{aligned}$$

In the following, we restrict ourselves to the lowest-order discretization, which means that $w_h \in \mathcal{W}_h$ is piecewise constant on elements, and therefore Problem 2 reduces to

$$(3.6) \quad \int_{\Omega} \phi \frac{\partial S_{\alpha,h}}{\partial t} w_h \, d\Omega + \sum_{f \in \mathcal{F}_h} \int_f (\mathbf{u}_{\alpha,h}(S_h, \mathbf{u}_h) \cdot \mathbf{n}_e)^* [w_h] \, dS = \int_{\Omega} \frac{q_{\alpha,h}}{\rho_{\alpha}} w_h \, d\Omega.$$

In matrix form, we have

$$(3.7) \quad \phi W \frac{\partial S_{\alpha,h}}{\partial t} = F(S_h, \mathbf{u}_h) + \frac{1}{\rho_{\alpha}} W q_{\alpha,h},$$

where

$$(3.8) \quad F(S_h, \mathbf{u}_h) = \sum_{f \in \mathcal{F}_h} \int_f (\mathbf{u}_{\alpha,h}(S_h, \mathbf{u}_h) \cdot \mathbf{n}_e)^* [w_h] \, dS,$$

and W is a mass matrix corresponding to the L^2 -inner products for functions in \mathcal{W}_h . In the lowest-order case, W reduces to a diagonal matrix whose entry (i, i) represents the volume of element i .

3.2.1. Choice of the numerical fluxes. In this section, we describe how the numerical fluxes in (3.6) are computed.

To this aim we need to evaluate the restriction of the phase velocity to the faces of the elements in the mesh. The phase velocity is defined by the following expressions:

$$(3.9) \quad \mathbf{u}_{\alpha,h} = f_{\alpha,h} (\mathbf{u}_h + \mathbf{G}_{\alpha,h}), \quad \mathbf{G}_{\alpha,h} = \sum_{\substack{\beta=o,w,g \\ \beta \neq \alpha}} \lambda_{\beta,h} (\rho_\alpha - \rho_\beta) \mathbf{K}_h g \nabla z,$$

where $\lambda_{\beta,h}$, ρ_α , \mathbf{K}_h , and $f_{\alpha,h}$ are piecewise constants associated with each element. Because of the fluid incompressibility assumption, ρ_α and ρ_β are constant over the domain, and therefore their restrictions to the faces of the mesh are trivially defined. To evaluate the permeability on a face f we follow the approach presented in [17, p. 134] and compute a componentwise harmonic mean of the values of the permeability tensor coming from element $+$ and element $-$ sharing face f :

$$(3.10) \quad (\mathbf{K}_h)_{i,j}^f = \frac{2(\mathbf{K}_h)_{i,j}^+ (\mathbf{K}_h)_{i,j}^-}{(\mathbf{K}_h)_{i,j}^+ + (\mathbf{K}_h)_{i,j}^-},$$

where i, j denote the components in the tensor. However, this procedure is only an approximation and may impact the gravity term (i.e., the vertical component) for distorted, sloping elements and full tensor coefficients. Other approaches could also be used to approximate the restriction of \mathbf{K} to a face of the mesh, such as upwinding \mathbf{K} in the direction of the phase velocity (see [8]).

The mobility $\lambda_{\beta,h}$ and the fractional flow function $f_{\alpha,h}$ in (3.9), which are part of the flux in (3.8), are approximated when reconstructing the flux across faces using an upwind method. The numerical fluxes are upwinded in the normal direction of the interfaces according to the flow direction of the individual phases:

$$(3.11) \quad \left(\mathbf{u}_{\alpha,h}(S_h, \mathbf{u}_h) \cdot \mathbf{n} \right)^* = \begin{cases} \mathbf{u}_{\alpha,h}(S_h^-, \mathbf{u}_h^-) \cdot \mathbf{n} & \text{if } (\mathbf{u}_{\alpha,h} \cdot \mathbf{n})^* \geq 0, \\ \mathbf{u}_{\alpha,h}(S_h^+, \mathbf{u}_h^+) \cdot \mathbf{n} & \text{if } (\mathbf{u}_{\alpha,h} \cdot \mathbf{n})^* < 0. \end{cases}$$

Without upwinding, the numerical solution may display oscillations, overshoots, or undershoots (e.g., saturations less than zero or greater than one) or converge to an incorrect solution [9, p. 163].

Equation (3.11) only implicitly defines the numerical fluxes $\mathbf{u}_{\alpha,h}^*$, since the upwind direction depends on both the total velocity and the gravitational term $\mathbf{G}_{\alpha,h}$. Without gravity the direction of the phase velocity is the same as the direction of the total velocity, i.e.,

$$(3.12) \quad \left(\mathbf{u}_{\alpha,h}(S_h, \mathbf{u}_h) \cdot \mathbf{n} \right)^* = \begin{cases} \mathbf{u}_{\alpha,h}(S_h^-, \mathbf{u}_h^-) \cdot \mathbf{n} & \text{if } (\mathbf{u}_h \cdot \mathbf{n}) \geq 0, \\ \mathbf{u}_{\alpha,h}(S_h^+, \mathbf{u}_h^+) \cdot \mathbf{n} & \text{if } (\mathbf{u}_h \cdot \mathbf{n}) < 0. \end{cases}$$

When accounting for gravity, the $\mathbf{G}_{\alpha,h}$ term also needs to be considered when finding the upwind direction. Hence determining the direction of the phase velocities is non-trivial, since $\mathbf{G}_{\alpha,h}$ actually contains one of the properties, $\lambda_{\beta,h}$, that we are trying to approximate in the upwinding. In our simulator, to determine phase velocity directions and upwinding, we use the approach proposed in [46] (see also [45]), which uses a heuristic approach to determine the upwind direction for $\mathbf{G}_{\alpha,h}$, and then check for consistency of the phase velocities $\mathbf{u}_{\alpha,h}$ and the upwind direction used for $\mathbf{G}_{\alpha,h}$.

3.3. Temporal discretization. Only the conservation law in (3.7) has an explicit time derivative dependence. The discretized system of equations is solved using an IMplicit Pressure Explicit Saturations (IMPES)-type method or—more accurately for a mixed system—IMplicit Pressure and Velocity Explicit Saturations. Specifically an improved IMPES-type method [16] is employed, where subtime stepping is used for saturations. Improved IMPES can be written as a fractional step time-advancing technique, where a large time step ΔT is used to update the total velocity and pressure unknowns, and a smaller time step $\Delta t = \frac{\Delta T}{k}$ is used to update the saturation unknowns. By denoting with $S^{n+\frac{i}{k}}$ the saturations at time $t = n\Delta T + i\Delta t$, we write the fractional step saturation update as

$$(3.13) \quad S_{\alpha,h}^{n+\frac{i+1}{k}} = S_{\alpha,h}^{n+\frac{i}{k}} + \Delta t \frac{1}{\phi} W^{-1} \left(F(S_h^{n+\frac{i}{k}}, \mathbf{u}_h^n) + \frac{1}{\rho_\alpha} W q_{\alpha,h}(S_h^{n+\frac{i}{k}}, p_h^n) \right),$$

where \mathbf{u}_h^n and p_h^n are the velocity and pressure at time $t = n\Delta T$ computed by solving Problem 3.

A pseudocode for the implementation is listed in Algorithm 1.

Algorithm 1 Pseudocode for improved IMPES implementation.

```

1: while  $t < t_{\text{final}}$  do
2:   Assemble the mixed system in Problem 3 given current saturations  $S_h^n$ 
3:   Solve the mixed system (3.2) for pressure  $\mathbf{p}_h^n$  and total velocity  $\mathbf{u}_h^n$ 
4:   Choose  $\Delta t$  based on the CFL condition in (3.15) and let  $k = \frac{\Delta T}{\Delta t}$ 
5:   for  $0, \dots, k-1$  do
6:     Compute  $S_h^{n+\frac{i+1}{k}}$  with (3.13)
7:      $t = t + \Delta t$ 
8:   end for
9: end while

```

Here $q_{\alpha,h}$ is computed as

$$(3.14) \quad q_{\alpha,h}(S_h, p_h) = \begin{cases} \text{rate} & \text{if injector,} \\ \rho_\alpha f_{\alpha,h}(S_h) q_h(p_h) & \text{if producer,} \end{cases}$$

where $q_h(p_h)$ is the total injection/production rate defined by (2.5) and *rate* is a user-supplied input.

To ensure the stability of the discretization, we choose Δt such that

$$(3.15) \quad \Delta t \leq \frac{c}{\phi} \min \left(\frac{h}{\left\| \frac{\partial \mathcal{F}(S, \mathbf{u})}{\partial S} \right\|_\infty}, \frac{\rho_\alpha}{\left\| \frac{\partial q_\alpha(S, \mathbf{u})}{\partial S} \right\|_\infty} \right),$$

where c is a user-supplied real number between 0 and 1 (in our tests $c = 0.9$). This choice is motivated by the fact that the first constraint for the time step is the CFL condition for the pure transport equation

$$(3.16) \quad \phi \frac{\partial S}{\partial t} = \nabla \cdot \mathcal{F}(S, \mathbf{u}),$$

and the second term is dictated by the stability region of forward Euler for the ODE

$$(3.17) \quad \phi \frac{\partial S}{\partial t} = \frac{1}{\rho_\alpha} q_\alpha(S, p).$$

4. Element-based algebraic multigrid (AMGe). AMGe is a framework for multigrid methods tailored to systems stemming from finite element discretizations. The components in AMGe are constructed from local element information, such as finite element matrices and element topology. This is in contrast to the classical AMG, where only system coefficients are used to construct the hierarchy of coarse spaces. It should be noted that the AMGe framework is significantly different from (the classical) AMG. Specifically, with this particular version of AMGe, we have guaranteed approximation properties, where a good convergent two-level AMG only has a weak approximation property (as a necessary condition; cf. the survey [56]). The latter deteriorates as the coarsening ratio (H/h) increases. This means that in the coarsening procedure, AMGe does not depend on a good heuristic, as used in AMG. That said, the choice of agglomerated elements does affect the resulting coarse spaces, and in the numerical results in section 5, it is demonstrated that taking into account the mesh anisotropy in the construction of the coarse agglomerated meshes improves the accuracy of the coarse systems.

Following the approach first described in [36, 38], an overview of the techniques involved for this specific version of AMGe are described in this section. For a more detailed description and for fundamental theory of the properties of the method, see [36] and [49]. The technique introduced in [36] extends the approach described in [49] and guarantees approximation properties of the coarse spaces for general unstructured meshes. The improved approximation properties of the coarse velocity spaces are achieved by introducing additional degrees of freedom associated with nonplanar interfaces between agglomerates. This leads to coarse spaces with the same stability and approximation properties as the original (fine-grid) Raviart–Thomas space.

4.1. Constructing agglomerates. Agglomerates are formed by grouping together fine-grid elements. Different techniques are available in our framework: graph partitioning techniques, octrees, geometric (coordinate-based) mesh partitioners, and other techniques that exploit directly the Cartesian or refinement structure of the mesh. To construct agglomerated elements using graph partitioning techniques, in particular, we build the dual graph of the mesh, which is an undirected graph, where each node of the graph represents an element in the mesh and node i is connected to node j if element i and element j share a face. METIS [32] is used for the partitioning of the undirected graph resulting in agglomerates consisting of fine-grid elements. Weights of the nodes and links in the dual graph (i.e., for the elements and faces of the mesh) can be provided to the partitioners in order to generate smaller agglomerated elements in parts of the domain or to modify the aspect ratio of coarse elements (see section 5.2). The coarse faces consist of the fine faces belonging to the intersection of any pair of neighboring agglomerates. This means that for unstructured meshes the coarse faces are nonplanar in most cases. The tentative number of fine-grid elements per agglomerate is a user-given input. A possible way to further improve the quality of the agglomerated meshes is by using problem-dependent weights for the graph partitioners [47].

It was found that straightforward graph partitioning algorithms sometimes produce agglomerates with “bad” or undesired topological properties, such as tunnels or holes. Breaking up the “bad” coarse elements into smaller agglomerates is then necessary in order to guarantee the well-posedness of the local problems involved in the computation of the coarse spaces (see section 4.3).

4.2. Building coarse pressure spaces. The coarse pressure spaces are constructed in the same way as introduced in [49], where the coarse space consists of

piecewise constant functions on agglomerated elements. We define the coarse pressure space $\mathcal{W}_H \subset \mathcal{W}_h$ to consist of functions which are constant on each agglomerated element. In addition, one can enrich the space \mathcal{W}_H by restricting additional functions, such as (higher-order) polynomial functions or other functions of interest. It is worth noticing that the coarse pressure space does not need to be conforming across agglomerated element interfaces. A basis of the space \mathcal{W}_H is then used to form the columns of the prolongation matrix $\mathbf{P}_p : \mathcal{W}_H \rightarrow \mathcal{W}_h$. In the following section, we will demonstrate how to construct a coarse velocity space $\mathcal{R}_H \subset \mathcal{R}_h$ such that $\nabla \cdot \mathcal{R}_H = \mathcal{W}_H$. This property is necessary to preserve the stability of the upscaled discretization and to guarantee that the spaces $(\mathcal{R}_H, \mathcal{W}_H)$ are inf-sup compatible.

It should be noted that in our coarsening procedure defined in section 4.5, p_H is a Galerkin projection of p_h via the solution of a coarse system (in the linear case).

Furthermore, we will assume that also the saturations are upscaled using the same coarse space \mathcal{W}_H . This is consistent with the choice of finite element spaces for the fine-grid discretization, but one could use different coarse spaces for upscaling the saturations.

4.3. Building coarse velocity spaces. The method used to create coarse finite element spaces for the lowest-order Raviart–Thomas space is described here. The explanation closely follows that of [36]. The method is a two-step process, where we first find the coarse basis functions on coarse faces and then extend the basis functions into the interior of the neighboring agglomerated elements.

The coarse basis functions are defined in terms of their fine degrees of freedom. Given a sufficiently smooth vector function \mathbf{r} and a fine face f , the value of the degree of freedom associated with f is defined as

$$(4.1) \quad \text{DoF}_f(\mathbf{r}) = \int_f \mathbf{r} \cdot \mathbf{n}_f dA,$$

where \mathbf{n}_f is the unit normal to the fine face and A is the surface area of the face.

For each coarse face F , a matrix \mathbf{W}_F is formed. \mathbf{W}_F consists of the values of the degrees of freedom of the fine faces $f_1, \dots, f_{|F|}$ constituting F :

$$(4.2) \quad \mathbf{W}_F = \begin{bmatrix} \text{DoF}_{f_1}(\mathbf{e}_1) & \text{DoF}_{f_1}(\mathbf{e}_2) & \text{DoF}_{f_1}(\mathbf{e}_3) & \text{sgn}(f_1, F) \int_{f_1} dA \\ \vdots & \vdots & \vdots & \vdots \\ \text{DoF}_{f_{|F|}}(\mathbf{e}_1) & \text{DoF}_{f_{|F|}}(\mathbf{e}_2) & \text{DoF}_{f_{|F|}}(\mathbf{e}_3) & \text{sgn}(f_{|F|}, F) \int_{f_{|F|}} dA \end{bmatrix},$$

where $|F|$ is the number of fine faces in the coarse face F . Here $\text{sgn}(f, F) = 1$ if the orientation of the fine face is equal to that of the coarse face, and $\text{sgn}(f, F) = -1$ otherwise. Above, \mathbf{e}_i stand for the three coordinate constant vector functions. The goal is to ensure that the coarse Raviart–Thomas space contains locally (on each agglomerated element) these constant vectors and hence have first order of approximation in L^2 as the fine-grid Raviart–Thomas space. The above construction is fairly general; we can include any given velocity functions of our interest in the coarse velocity space and maintain the compatibility; i.e., the coarse pressure space should contain their divergence.

Using an SVD $\mathbf{W}_F = \mathbf{U}\Sigma\mathbf{V}^T$, the linearly dependent columns of \mathbf{W}_F are eliminated. It is worth noting that this SVD computation scales linearly with the number of fine degrees of freedom which belongs to the coarse face. This is due to the fact that (1) we perform the *thin* SVD [26] and that (2) the number of columns in \mathbf{W}_F

is small and independent of the number of fine degrees of freedom. The left singular vectors (columns of \mathbf{U}) \mathbf{u}_j are chosen based on the corresponding singular values σ_j . If $\sigma_j \geq \epsilon \sigma_{\max}$, where $\epsilon \in (0, 1]$ is a user-given input, then \mathbf{u}_j defines a coarse basis function for the coarse face F denoted by \mathbf{r}_F^j . More precisely, \mathbf{r}_F^j is only equal to \mathbf{u}_j on the coarse face F and zero everywhere else. If the coarse face F is planar, then only σ_1 would be different from 0 and only one coarse trace will be selected; for a nonplanar face up to four coarse traces will be selected depending on the tolerance ϵ . This is in contrast with multilevel multiscale mimetic methods (M^3) [41, 40], where only one coarse degree of freedom is used for each face.

The above procedure describes the first step in finding a coarse velocity space. The second step involves taking the partially defined functions \mathbf{r}_F^j and extending these coarse basis functions into the interior of the coarse elements. The extension is performed using the approach in [49], which guarantees that the divergence of the coarse Raviart–Thomas space belongs to the coarse L^2 space. More specifically, for each coarse element T we define the local finite element spaces:

$$\begin{aligned}\tilde{\mathcal{R}}_T &= \{\mathbf{v}_h \in \mathcal{R}_h \mid \text{supp}(\mathbf{v}_h) \subset T \text{ and } \mathbf{v}_h \cdot \mathbf{n} = 0 \text{ on } \partial T\} \\ \text{and } \tilde{\mathcal{W}}_T &= \{w_h \in \mathcal{W}_h \mid \text{supp}(w_h) \subset T \text{ and } (w_h, 1) = 0\}.\end{aligned}$$

Given a partially defined function \mathbf{r}_F^j on the coarse face F belonging to the coarse element T , the local (element-based) mixed system reads as follows.

PROBLEM 4 Find $(\mathring{\mathbf{r}}_T, p_h) \in \tilde{\mathcal{R}}_T \times \tilde{\mathcal{W}}_T$ such that

$$\begin{cases} \left(\alpha (\mathring{\mathbf{r}}_T + \mathbf{r}_F^j), \mathbf{v}_h \right)_T + (p_h, \nabla \cdot \mathbf{v}_h)_T = 0 & \forall \mathbf{v}_h \in \tilde{\mathcal{R}}_T, \\ \left(\nabla \cdot (\mathring{\mathbf{r}}_T + \mathbf{r}_F^j), w_h \right)_T = 0 & \forall w_h \in \tilde{\mathcal{W}}_T. \end{cases}$$

The coefficient matrix α (a 3×3 symmetric positive definite (SPD) matrix) can be set equal to the coefficients from the original problem \mathbf{K}^{-1} , but this is not strictly necessary. Problem 4 is guaranteed to have a unique solution [36, 49]. By solving these local problems on each pair of agglomerates (T^+, T^-) adjacent to a coarse face F , we obtain the coarse basis functions $\mathbf{r}_h = \mathbf{r}_F^j + \mathring{\mathbf{r}}_{T^+} + \mathring{\mathbf{r}}_{T^-}$ of the space \mathcal{R}_H . We finally let the columns of the prolongation matrix $P_{\mathbf{u}} : \mathcal{R}_H \rightarrow \mathcal{R}_h$ be the collection of the coarse basis functions \mathbf{r}_h .

4.4. Dealing with localized sources. Localized or pointwise source terms, such as the well's terms in (3), represent an additional challenge for many upscaling techniques. In fact, localized source terms may drastically reduce the accuracy of traditional upscaling techniques based on homogenization, since the exact location of the source is lost on the coarser mesh.

In contrast, the AMGe approach offers great flexibility in dealing with localized sources and it allows for accurate upscaling of localized sources. In particular, two different approaches are possible. One option is to use smaller agglomerated elements in proximity of a localized source. This can be achieved by simply leaving some fine elements in the neighborhood of the localized source unagglomerated or by using a weighted graph partitioning algorithm. This approach is the equivalent of adaptive mesh refinement (h -refinement) in the finite element settings. The other option is

to locally enrich the coarse space by adding additional functions with support in a neighborhood of the localized sources, as described above. This approach is similar in spirit to p -refinement in the finite element settings. Actually, in our case these approaches should be viewed as *adaptive coarsening*. In the numerical results presented in this paper we use the first approach to locally refine the agglomerated mesh in the neighborhood of a localized source. Figure 12(c) provides an example of an agglomerated mesh where the elements which contain wells and their immediate neighbor cells are left unagglomerated.

4.5. Upscaling with AMGe. The construction of the coarse spaces and thereby the interpolation operators is done in a setup phase, while the rest of the computation is performed entirely on the coarse agglomerated meshes. The fine grid is visited only when the solution is prolonged back to the fine grid for visualization purposes. For some applications, where there is no need for the solution on the fine grid, but only scalar quantities—such as production data—are important, the fine grid is therefore touched only once (in the setup phase) and not during the rest of the simulation.

The upscaled mixed system

$$(4.3) \quad \begin{bmatrix} M & B^T \\ B & -C \end{bmatrix}_H \begin{bmatrix} \mathbf{U} \\ P \end{bmatrix}_H = \begin{bmatrix} \mathbf{F}_u \\ F_p \end{bmatrix}_H$$

is assembled directly for the upscaled spaces. The local coarse matrices (one for each agglomerated element) are precomputed in the setup phase and then assembled into the upscaled system at each time step without visiting the fine grid. By construction, the upscaled mixed system assembled on the coarse spaces is equivalent to the following Galerkin projection:

$$(4.4) \quad \begin{bmatrix} M & B^T \\ B & -C \end{bmatrix}_H = \begin{bmatrix} P_u^T & 0 \\ 0 & P_p^T \end{bmatrix} \begin{bmatrix} M & B^T \\ B & -C \end{bmatrix}_h \begin{bmatrix} P_u & 0 \\ 0 & P_p \end{bmatrix} \text{ and } \begin{bmatrix} \mathbf{F}_u \\ F_p \end{bmatrix}_H = \begin{bmatrix} P_u^T & 0 \\ 0 & P_p^T \end{bmatrix} \begin{bmatrix} \mathbf{F}_u \\ F_p \end{bmatrix}_h,$$

where H and h , in this case, respectively represent the upscaled level and the fine-grid level. The strong conservation properties of the finite element spaces $(\mathcal{R}_h, \mathcal{W}_h)$, the AMGe coarsening technique that ensures $\nabla \cdot \mathcal{R}_H = \mathcal{W}_H$ (described in section 4.3), and the special treatment of the wells (described in section 4.5) guarantee a strong mass conservation for the upscaled system. In particular, when we leave wells unagglomerated, we have that $f_h = P_p f_H$, and therefore $(\nabla \cdot u_H - C_H p_H, w_H) = (f_H, w_H)$ implies $(\nabla \cdot P_u u_H - C_h P_p p_H, w_h) = (f_h, w_h)$.

In a way similar to the mixed system, the upscaled saturation equations read as

$$\int_{\Omega} \phi \frac{\partial S_{\alpha,H}}{\partial t} w_H d\Omega + \sum_{F \in \mathcal{F}_H} \sum_{f \in \mathcal{F}} \int_f (\mathbf{u}_{\alpha,H}(S_H, \mathbf{u}_H) \cdot \mathbf{n})^* [w_H] dS = \int_{\Omega} \frac{q_{\alpha,H}}{\rho_{\alpha}} w_H d\Omega,$$

where \mathcal{F}_H represents the set of coarse faces F in the agglomerated set. It is worth noticing that for the choice of piecewise constant saturations on agglomerated elements the local matrices representing the integral $\sum_{f \in \mathcal{F}} \int_f (\mathbf{u}_{\alpha,H}(S_H, \mathbf{u}_H) \cdot \mathbf{n})^* [w_H] dS$ can be precomputed in the setup phase for each coarse degree of freedom of the coarse total velocity \mathbf{u}_H . Therefore, to evolve the saturation equations in time is not necessary to visit the fine mesh during the simulation.

In the numerical results presented in the following section, we use MINRES preconditioned by a block-diagonal AMG preconditioner to solve the mixed system involving total velocity and pressure (see [43] or [55]). The upscaled coarse-grid problems are solved with a sparse direct solver. For large-scale problems, the coarse-grid

TABLE 1
Input parameters.

Porosity ϕ	0.3	-
Viscosity oil μ_o	1.14	cP
Viscosity water μ_w	0.096	cP
Density oil ρ_o	800	kg/m ³
Density water ρ_w	1022	kg/m ³

problems should also be solved with an iterative solver. However, the coarse-grid mixed systems can be more ill-conditioned than the original fine-grid system due to the complicated geometry of the agglomerates and the possibly nonuniform distribution of coarse degrees of freedom within each agglomerated element. This issue is not addressed in the present paper, and we refer the reader to [18, 19], where we present our progress on the construction of efficient solvers for the upscaled system using iterative methods.

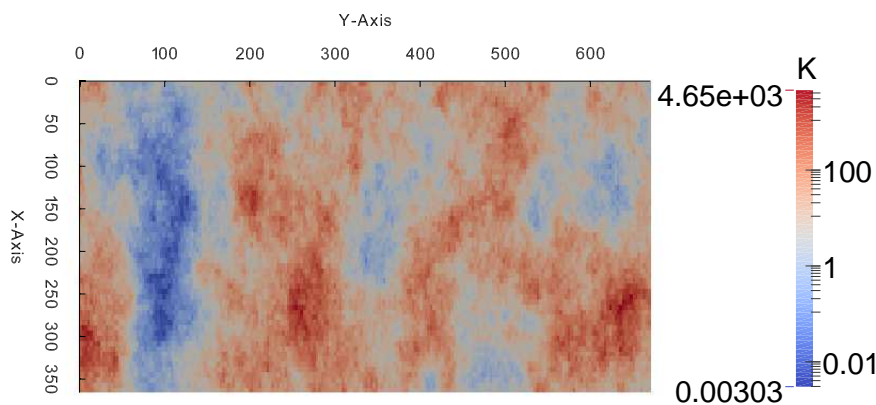
5. Numerical results. Two numerical experiments are carried out to test the accuracy of the numerical upscaling. We consider the 10th SPE Comparative Solution Project (SPE10 dataset 2) top layer [51] and a three-dimensional (3D) model derived from the SAIGUP model where some features (i.e., faults) were not included. The SPE10 model has a regularly structured mesh with a highly heterogeneous permeability field. The SAIGUP model has a more challenging and realistic geometry. Simulations are carried out using our numerical upscaling technique, and the results are compared to the simulation results from the fine-grid reference model. For all simulations, the input values given in Table 1 are used.

The aim of the numerical results presented in this section is to demonstrate on nontrivial benchmarks the efficiency—in terms of reduction of the computational cost for the same accuracy requirement—of our mathematically rigorous approach to numerical upscaling. For a comparison with traditional upscaling techniques we refer the reader to [18], where we compared our approach with the flow-based upscaling method in the commercial software Petrel and demonstrated superior accuracy for the same dimension of the upscaled problems in a 3D benchmark derived from the SPE10 model.

The software developed in this work uses the finite element library MFEM [44] from Lawrence Livermore National Laboratory (LLNL). MFEM is a general, modular, parallel C++ library for FEM research and development. It supports a wide variety of finite element spaces in two and three dimensions, as well as many bilinear and linear forms defined on them. It includes classes for dealing with various types of triangular, quadrilateral, tetrahedral, and hexahedral meshes and their global and local refinement. Parallelization in MFEM is based on MPI, and it leads to high scalability in the finite element assembly procedure. It supports several solvers from the hypre library [28].

5.1. SPE10 top layer. As a first example, a test case using the top layer of the SPE10 x-permeability field is studied [51]. A two-dimensional (2D) model is chosen as a first case to allow for better analyses of the accuracy obtained in the upscaling procedure. Figure 2 displays the permeability field. We simulate the fine-grid reference model and compare this to three upscaled models with different levels of coarsening.

The mesh is a regularly structured grid with $60 \times 220 \times 1$ elements, where each

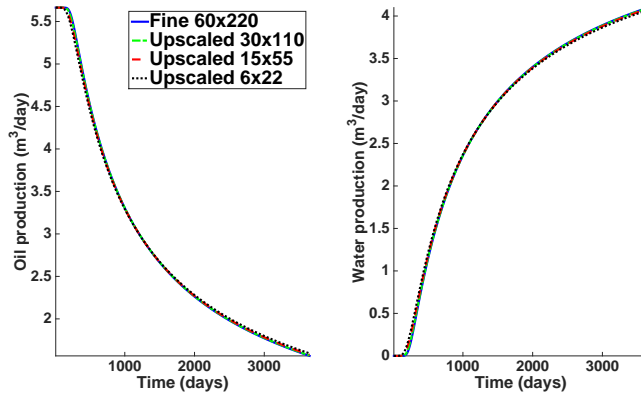
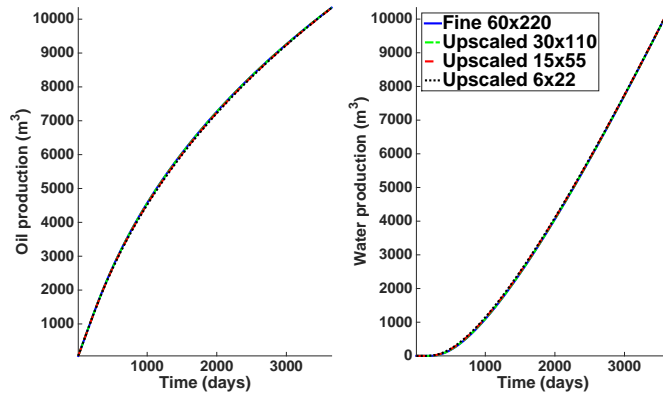
FIG. 2. SPE10 x -permeability for the top layer.TABLE 2
Degrees of freedom, number of nonzeros, and complexities.

Problem	#elements	#faces	#DoFs	nnz	Arithmetic complexity	Operator complexity
60×220 (fine)	13200	53080	92680	607485	-	-
30×110	3317	13418	23371	152965	1.25252	1.2518
15×55	842	3446	5986	39327	1.06491	1.06474
6×22	149	634	1093	7351	1.01199	1.0121

element has a size of $6.096 \times 3.048 \times 0.6096$ meters. A water injection well is placed in the middle of the mesh, and four producers are placed in the corner elements. The producers are controlled by a bottom hole pressure of 175 bars. The injection well is controlled by a constant injection rate of 0.5 times the element volume. All wells have a radius of 0.2 meters. The porosity is 0.3 for all elements. The whole reservoir has an initial oil saturation of 1.

Constant time steps of 10 days are used. The simulation horizon is 10 years. Time integration is carried out with the improved IMPES method, as explained in section 3.3. Since the main interest of this paper is on the numerical upscaling, we have purposely kept the time stepping very simple. Figures 3 and 4 show the daily and accumulated production (and accumulated injection) for both the fine-grid reference solution and the upscaled solutions. Structured coarsening is used, where, respectively, 4 (30×110), 16 (15×55), and 100 (6×22) fine-grid elements are grouped together into one agglomerate. Table 2 contains information about the number of degrees of freedom, the number of nonzeros, and complexities for the different levels of coarsening. The arithmetic complexity C_a is defined as the ratio of the total number of degrees of freedom on all levels (fine-grid and upscaled) to the fine-grid number of degrees of freedom. In a similar way, the operator complexity C_o is the ratio of the total number of nonzeros (in the mixed system) on all levels to the number of nonzeros on the fine grid. More specifically, we have

$$(5.1) \quad C_a = \frac{\sum_{l=0}^{\text{levels}-1} \dim(\mathcal{R}_{h(l)} \times \mathcal{W}_{h(l)})}{\dim(\mathcal{R}_{h(0)} \times \mathcal{W}_{h(0)}), \quad C_o = \frac{\sum_{l=0}^{\text{levels}-1} \text{nnz}(\mathbf{A}_{h(l)})}{\text{nnz}(\mathbf{A}_{h(0)})}.$$

FIG. 3. Daily production data for fine-grid 60×220 and upscaled models.FIG. 4. Accumulated production data for fine-grid 60×220 and upscaled models.

We stress the fact that many methods in practice can achieve C_a close to unity and have acceptable approximation properties. However, it is of vital practical importance to also ensure that C_o is close to unity (or at least sufficiently less than two), since then we can store the upscaled problem with memory (much) less than the original fine-grid problem.

As can be seen from Figures 3 and 4, the difference between the fine-grid reference solution and the upscaled solutions is small and even difficult to spot in these plots.

To get a better idea of the errors committed in the upscaling, Figures 5 and 6 show the difference between the upscaled and the fine-grid production relative to the fine-grid production/injection. Both daily and accumulated production error curves are shown. As evident from the figures, the error committed even for highly aggressive coarsening (100 fine elements per agglomerate) is less than 3%.

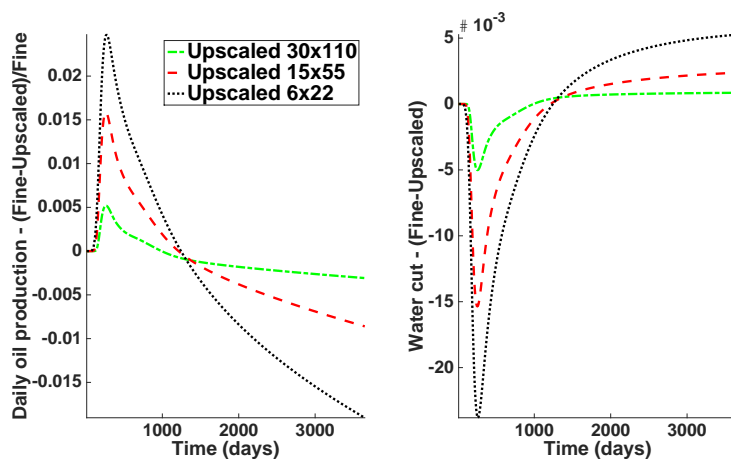


FIG. 5. Difference between fine-grid 60×220 and upscaled models in terms of daily oil production and water cut.

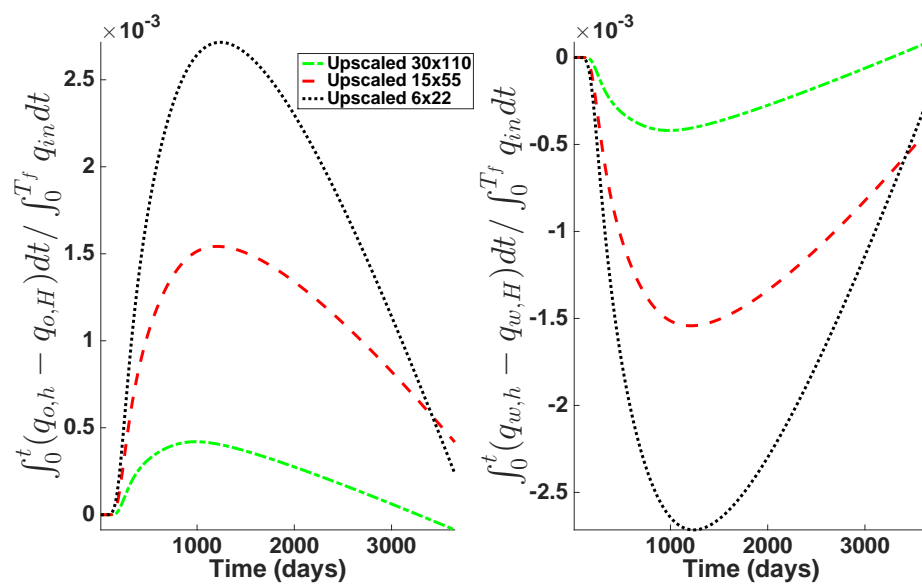


FIG. 6. Difference in production between the fine-grid and upscaled models integrated over time relative to the total injection of water over time.

TABLE 3
Number of days before water breakthrough for each well.

Well	Coarse problem size			
	60×220	30×110	15×55	6×22
Bottom right	210	200	170	160
Top right	490	470	410	330
Top left	810	820	830	810
Bottom left	680	680	650	610

By comparing Figures 3 and 5, it is clear that most of the error is at the time where water arrives at the production wells. It seems that the upscaled models predict slightly earlier water breakthrough compared to the fine-grid reference solution. This is to be expected, since the upscaling results in a more diffusive discretization, which means small amounts of water will be moved more quickly to the production wells. Mitigating this effect is an issue, which will be covered in future communications, e.g., by adaptively refining the coarse spaces in the neighborhood of the water front. Nevertheless, the error committed in the upscaling is very small and it is a very satisfying result for such a heterogeneous permeability field, where permeability values are varying by six orders of magnitude.

As previously mentioned, the upscaled solutions have a tendency to predict water breakthrough at an earlier time compared to the fine-grid reference solution. To better estimate exactly how much earlier, Table 3 contains the number of days before water breakthrough is predicted at each well for both the fine-grid solution and the upscaled solutions. In this table, water breakthrough is defined as the time where 1% of the produced fluids is water.

Clearly, the amount of coarsening has a large impact on the ability to accurately predict water breakthrough. The upscaled solutions seem to compensate for the early water breakthrough by later producing less water, ultimately causing the accumulated production to converge over time towards the same amount. This compensating behavior was also observed using other upscaling techniques that enforce strong mass conservation and in absence of capillary pressure (see, e.g., [58]).

In order to give a better picture of how the upscaled saturation profiles compare to the fine-grid saturation profile, the water saturation at the time of water breakthrough and after 10 years is plotted in Figures 7 and 8. Starting from the top row, the plots show the upscaled 6×22 , 15×55 , 30×110 together with the the fine-grid 60×220 water saturations. The absolute difference between the fine-grid reference solution and the upscaled solutions is plotted in the right column. The errors given in the captions are computed by interpolating the upscaled solution onto the fine grid, subtracting this from the fine-grid reference solution. and then computing the weighted $L^2(\Omega_h)$ -inner product.

5.1.1. Multilevel recursive upscaling. In this section, the multilevel recursive upscaling capability of the AMGe method is demonstrated for the top layer of the SPE10. Figure 9 shows the agglomeration of the fine-grid elements. METIS is used in a recursive way to find successive coarse levels of agglomerated elements. Five levels (level zero is the fine-grid mesh) are generated, where a coarsening factor of 16 is used from level zero to level one and a coarsening factor of four for the remaining agglomeration steps. Elements containing wells are left unagglomerated. Given this hierarchy of agglomerated elements, simulations are carried out on each level, and the resulting upscaled solutions are compared to the fine-grid reference solution.

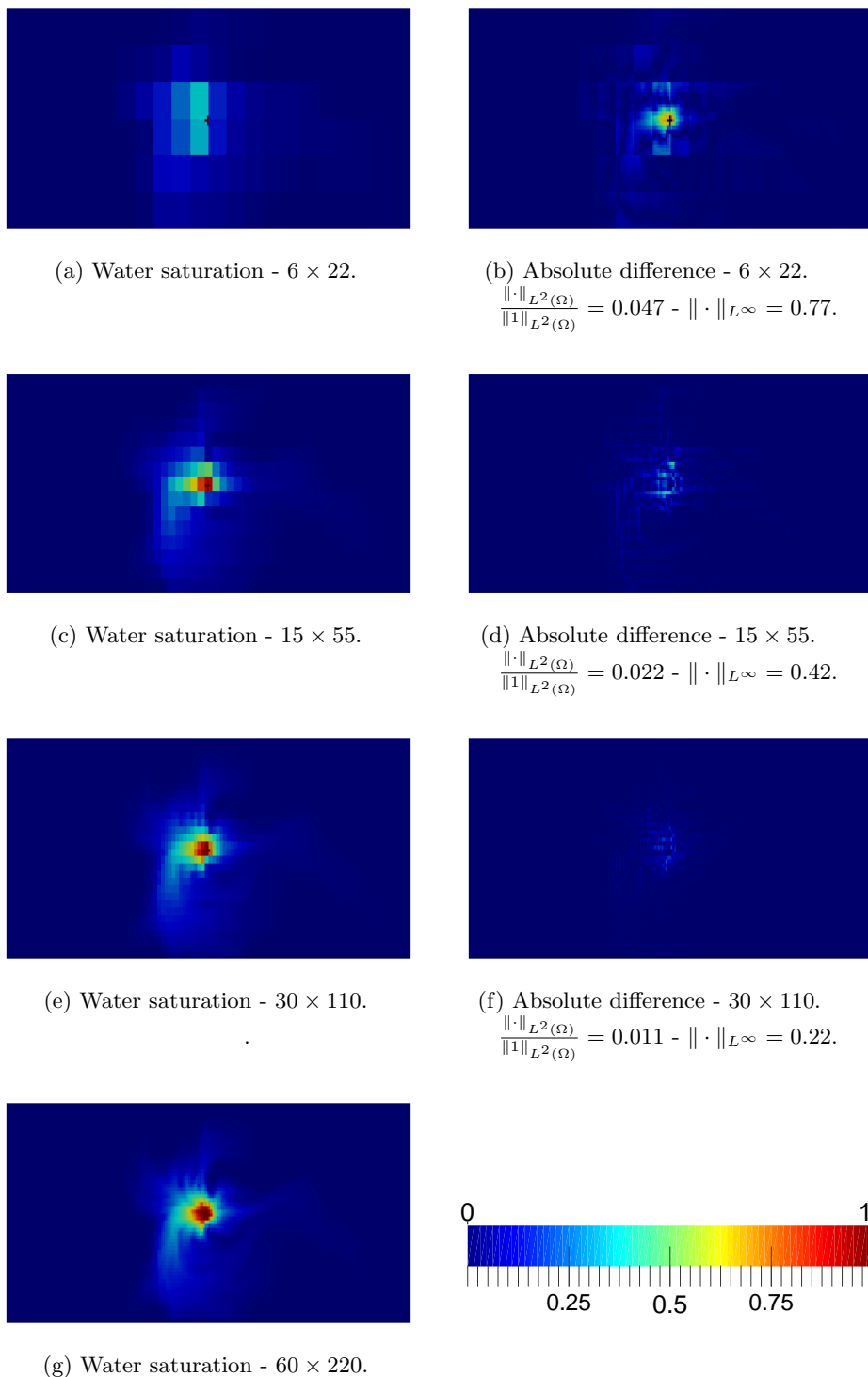


FIG. 7. Left column: Water saturation after first water breakthrough (after 210 days) for the fine-grid reference solution and the three upscaled solutions. Right column: Absolute difference between the fine-grid reference solution and the three upscaled solutions.

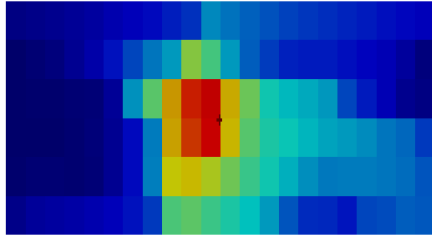
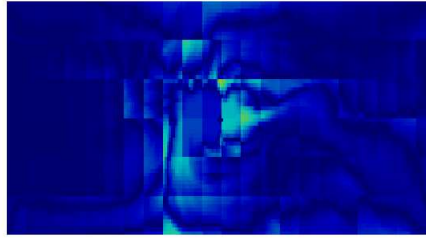
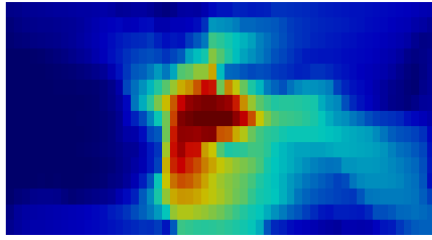
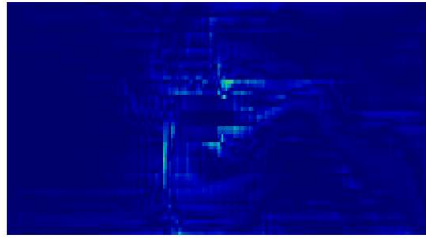
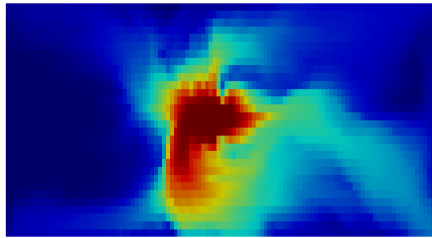
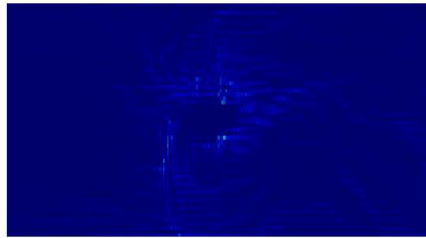
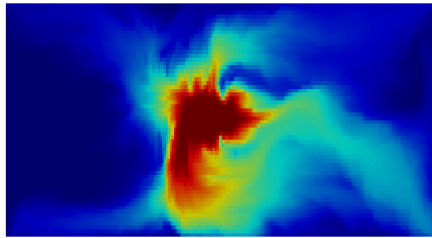
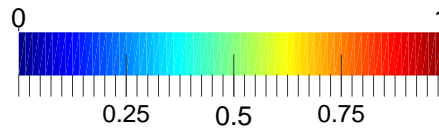
(a) Water saturation - 6×22 .(b) Absolute difference - 6×22 .
 $\frac{\|\cdot\|_{L^2(\Omega)}}{\|1\|_{L^2(\Omega)}} = 0.095 - \|\cdot\|_{L^\infty} = 0.60.$ (c) Water saturation - 15×55 .(d) Absolute difference - 15×55 .
 $\frac{\|\cdot\|_{L^2(\Omega)}}{\|1\|_{L^2(\Omega)}} = 0.048 - \|\cdot\|_{L^\infty} = 0.49.$ (e) Water saturation - 30×110 .(f) Absolute difference - 30×110 .
 $\frac{\|\cdot\|_{L^2(\Omega)}}{\|1\|_{L^2(\Omega)}} = 0.023 - \|\cdot\|_{L^\infty} = 0.34.$ (g) Water saturation - 60×220 .

FIG. 8. Left column: Water saturation after 10 years for the fine-grid reference solution and the three upscaled solutions. Right column: Absolute difference between the fine-grid reference solution and the three upscaled solutions.

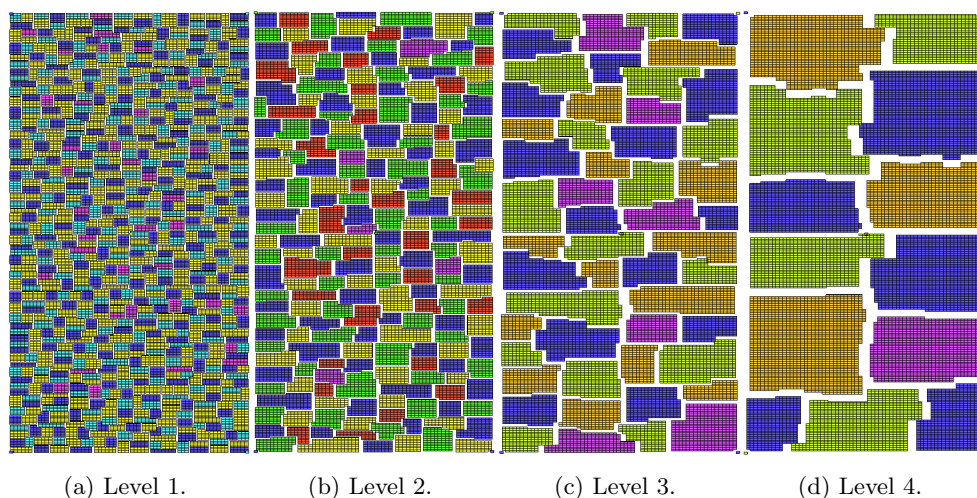


FIG. 9. Unstructured (METIS) recursive agglomeration of the SPE10 top layer. Elements containing wells are unagglomerated.

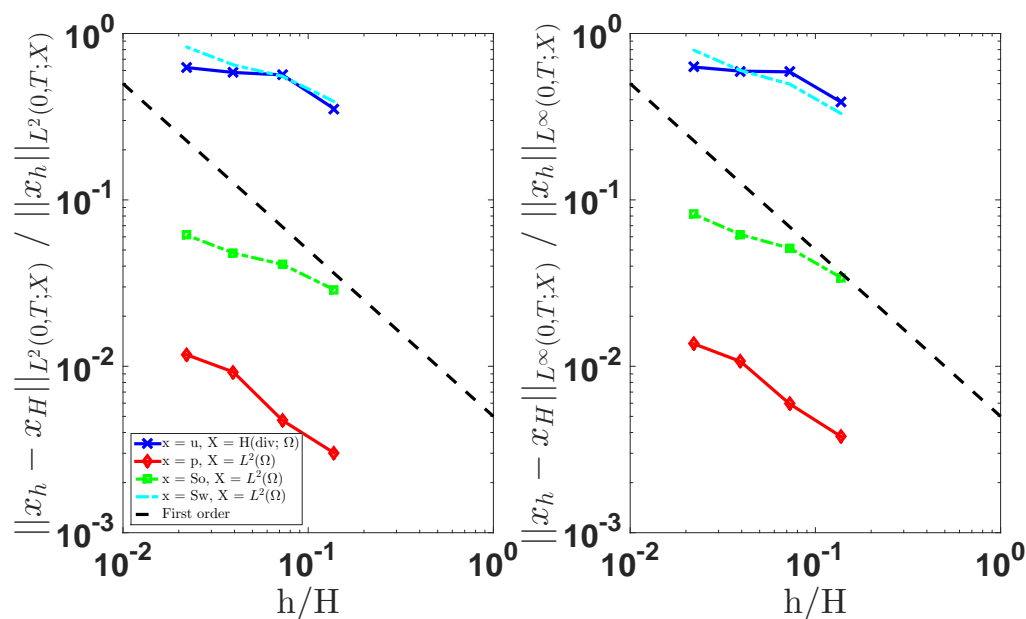


FIG. 10. Error norms (5.2) as a function of the coarsening ratio h/H , defined as the square root of the ratio between the number of degrees of freedom for the upscaled system and the number of degrees of freedom for the fine-grid system.

Figure 10 shows the error in the solution between the fine-grid reference solution (level zero) and the solutions for the four (levels 1, 2, 3, 4) upscaled problems as a function of h/H (approximated by the square root of the ratio of the number of degrees of freedom for the upscaled problem relative to the number of degrees of freedom for the fine-grid problem). The plots in Figure 10 show the $L^2(0, T; X)$ error

TABLE 4

Degrees of freedom and number of nonzeros for recursive upscaling of the top layer of SPE10.

Level	#DoFs	#DoFs(u)	#DoFs(p)	nnz
0 (fine)	92680	53080	13200	607485
1	9107	6614	831	107355
2	2529	1896	211	37213
3	746	575	57	13346
4	233	179	18	4240

norm (left) and $L^\infty(0, T; X)$ error norm (right), defined as

$$(5.2) \quad \begin{aligned} \|x_h - x_H\|_{L^2(0, T; X)} &= \sqrt{\int_0^T \|x_h(t) - x_H(t)\|_X^2 dt}, \\ \|x_h - x_H\|_{L^\infty(0, T; X)} &= \max_{t \in [0, T]} \|x_h(t) - x_H(t)\|_X. \end{aligned}$$

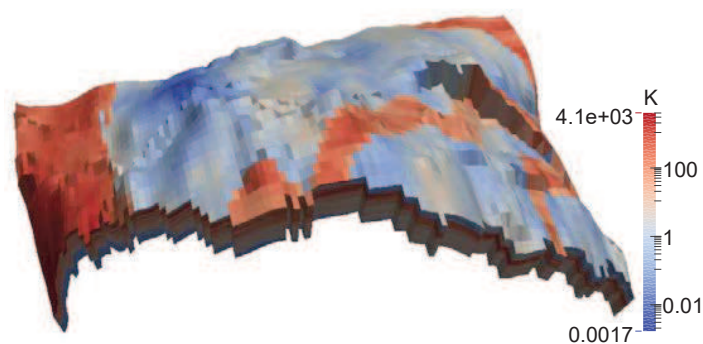
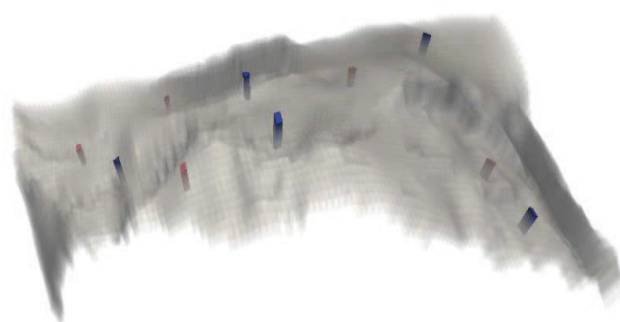
Here x_h and x_H are generic notation for the fine-grid unknowns $p_h, u_h, (S_o)_h, (S_w)_h$ and the upscaled unknowns $p_H, u_H, (S_o)_H, (S_w)_H$, respectively. Similarly, X is a generic notation for the functional spaces $H(\text{div}, \Omega), L^2(\Omega), L^2(\Omega), L^2(\Omega)$ where the unknowns are defined.

Both error norms are normalized by the appropriate norm of the fine-grid reference solution. The exact formula is given by the labels in the figure. the order of the discretization is also plotted as a base scenario for comparison. Figure 10 suggests that our upscaling approximates the solution very well: increasing errors as a function of coarsening is inevitable, since we are necessarily losing information, but the sublinear rate of this increase demonstrates that the AMGe coarse spaces lead to more accurate results than solving the problem using standard finite elements on an equally coarse grid (i.e., for the same number of degrees of freedom). Finally, the number of degrees of freedom and the number of nonzeros for the upscaled problems are given in Table 4. As evident from Table 4, the multilevel recursive upscaling provides a nice reduction in both the number of degrees of freedom and the number of nonzeros. In fact, the arithmetic complexity for the overall multilevel upscaling (all levels included) is 1.14, and similarly the operator complexity is 1.27.

5.2. Modified SAIGUP. The SAIGUP study is a project with the purpose to *quantify objectively the sensitivity of geological complexity on production forecasts, as a function of generic aspects of both the sedimentological architecture and faulted structure of shallow marine hydrocarbon reservoirs and to validate these results using real-case reservoir and production data* [52]. In this work, we use the same geometry and permeability field as in the SAIGUP benchmark in order to apply our upscaling techniques to a more realistic and geometrically challenging case than the SPE10.

The SAIGUP mesh contains faults, which—at present—we are not able to account for. For this reason, the original SAIGUP is remeshed to remove the faults in the reservoir. This is done by extracting the (x, y) coordinates for all the vertices and the z coordinates for the vertices of top and bottom surfaces and then creating the remaining z coordinates in between by interpolation so that the original number of layers in the mesh is obtained. This procedure results in the mesh illustrated in Figure 11(a) (plotted with the permeability field) and in Figure 11(b) (plotted with the wells). Note that, for visualization purposes, we rescaled the mesh to emphasize the vertical features of the geometry, but in reality it is much flatter.

The remeshing procedure results in cells with bad aspect ratio (~ 20), where the faults were originally located; however, aspect ratios in reservoir simulation grids can be even worse. The rest of the cells have aspect ratios around 6. This could be

(a) Permeability field K_x .

(b) Water injection wells are marked with red, and production wells are marked with blue.

FIG. 11. Modified SAIGUP permeability field and well locations (scaled with $0.5x$ in the y -direction and $6x$ in the z -direction).

remedied by cutting cells into two or more, but this has not been done for this work.

There are five production wells, which all are perforated in the top 14 layers. All production wells have a bottom hole pressure of 175 bars. There are five water injection wells, which all are perforated in the bottom 12 layers. All the perforations of the injection wells are set to inject 0.0058 times the element volume per day. This is a total injection of 7892 m^3 of water per day. The simulation horizon is 30 years with constant time step sizes of 30 days. This means that after 30 years we have injected roughly 15% of the pore volume ($\phi \cdot V = 0.3 \cdot 1.85 \text{ km}^3$) of the mesh. Since the main interest of the paper is on the numerical upscaling, we have purposely kept the time stepping very simple.

Unstructured (using METIS) and structured coarsening are applied to this mesh to form the agglomerates. Four different agglomeration strategies have been applied:

- *Full coarsening.* A graph partitioning algorithm is called directly on the element-to-element connectivity graph without any preprocessing to take into account wells (Figure 12(a)).
- *Full coarsening with wells.* Well elements and neighbor elements of wells are removed from the connectivity graph and left unagglomerated on the coarse mesh.
- *(x, y) -semicoarsening with wells.* We remove interelement connections in the graph if the normal of the shared face is almost vertical. In addition we do not agglomerate elements with wells and their neighbors (Figure 12(b)).
- *Structured (x, y) -semicoarsening with wells.* Utilizing the underlying Cartesian topology of the mesh, structured partitioning is used. This strategy is denoted “Cartesian semi” (Figure 12(c)).

Note that we use a coloring algorithm to show the agglomerated elements. Table 5 contains information about the number of degrees of freedom, the number of nonzeros, and complexities for the different types of coarsening.

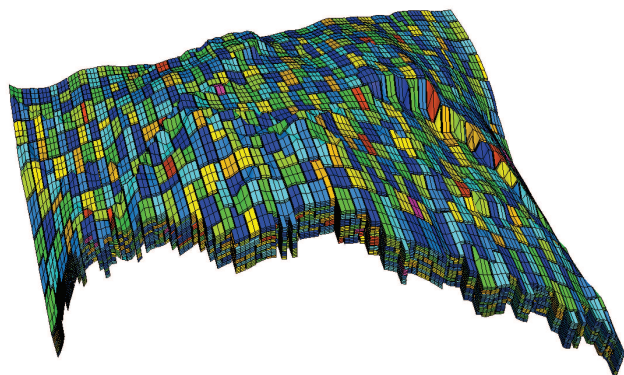
Figures 13(a) and 13(b) show the daily and accumulated production data for the fine-grid reference solution and the three different agglomeration strategies. The number in parentheses, (4), (16), or (64), indicates how many fine-grid elements are grouped into one agglomerate.

Figures 14(a) and 14(b) show the difference between the upscaled solutions and the fine-grid reference solution. Given the appropriate coarsening strategy, we are able to approximate the fine-grid reference solution with a very good accuracy. The choice of agglomeration strategy clearly has a big impact on the accuracy of the upscaled solution. By leaving the elements and neighbor elements of the wells unagglomerated, the upscaled solutions are more accurate. Furthermore, due to the strong coupling in the vertical direction, applying (x, y) -semicoarsening also provides an even more accurate upscaled solution. However, by looking at Table 5, we can see that the operator complexities are a bit high for the unstructured semicoarsening. By utilizing the underlying Cartesian topology of the mesh to generate a structured partitioning, the operator complexities drop significantly. This is due to two effects: on the one hand, with Cartesian agglomeration, coarse faces are fewer and tend to be more planar, leading to a reduction in the number of degrees of freedom (as can also be seen by the decrease in arithmetic complexity); on the other hand, each agglomerated element tends to have fewer coarse faces, resulting in smaller dense elemental matrices and therefore in a sparser global upscaled mixed system. This stresses the importance of a good quality of the agglomerated mesh. A general purpose software like METIS allows a lot of flexibility at the cost of a more expensive upscaled system. For this reason, we advocate whenever possible exploiting all the information of the fine-grid topology and problem parameters to improve both the accuracy and the computational efficiency of the upscaled model.

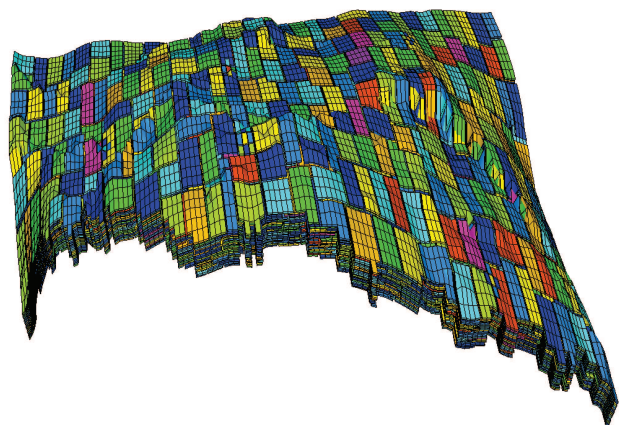
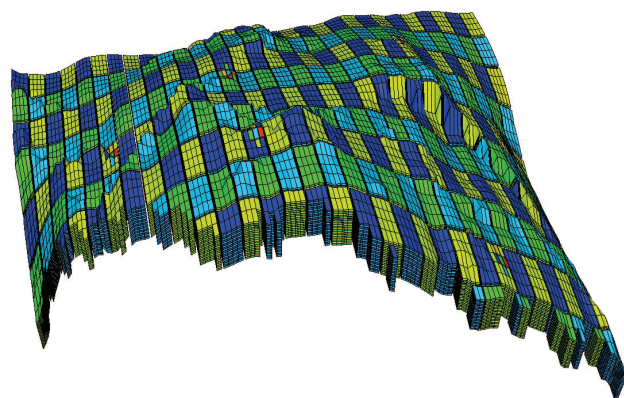
TABLE 5

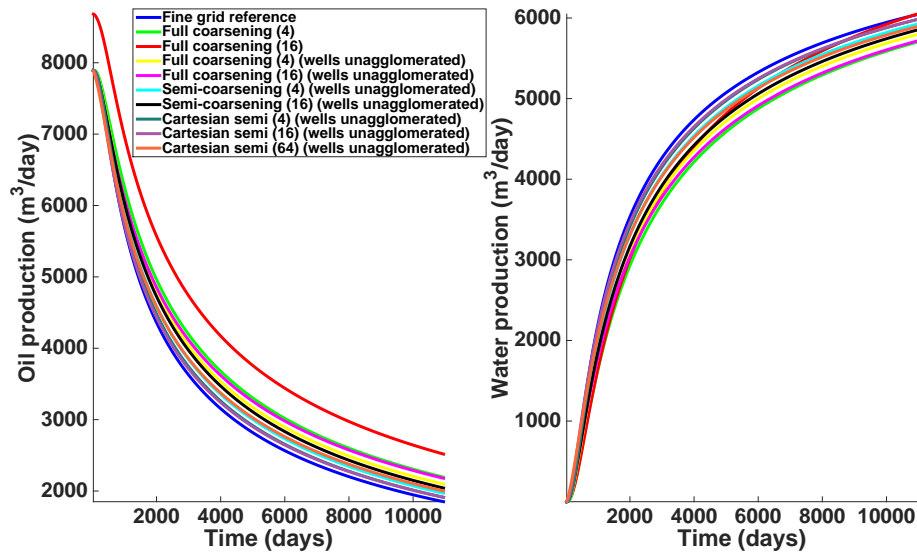
*Degrees of freedom, number of nonzeros, and complexities. * means elements with wells, and immediate neighbor elements of wells are left unagglomerated.*

Problem	#elements	#faces	#DoFs	nnz	Arithmetic complexity	Operator complexity
Fine grid	78720	243576	479736	3549946	-	-
Full coarsening (4)	17210	82081	168060	3039542	1.41412	1.85622
Full coarsening (16)	4920	25633	65422	1620930	1.17211	1.45661
Full coarsening (4)*	17960	84401	171960	3054520	1.4221	1.86044
Full coarsening (16)*	5616	28108	69924	1697050	1.18211	1.47805
Semicoarsening (4)*	14761	73982	173180	4080239	1.44573	2.14938
Semicoarsening (16)*	5629	30408	81513	2564226	1.21798	1.72233
Cartesian semi (4)*	20660	64372	175336	2194200	1.41582	1.61809
Cartesian semi (16)*	5970	19001	61025	901477	1.1523	1.25394
Cartesian semi (64)*	2100	6653	20683	318655	1.05114	1.08976

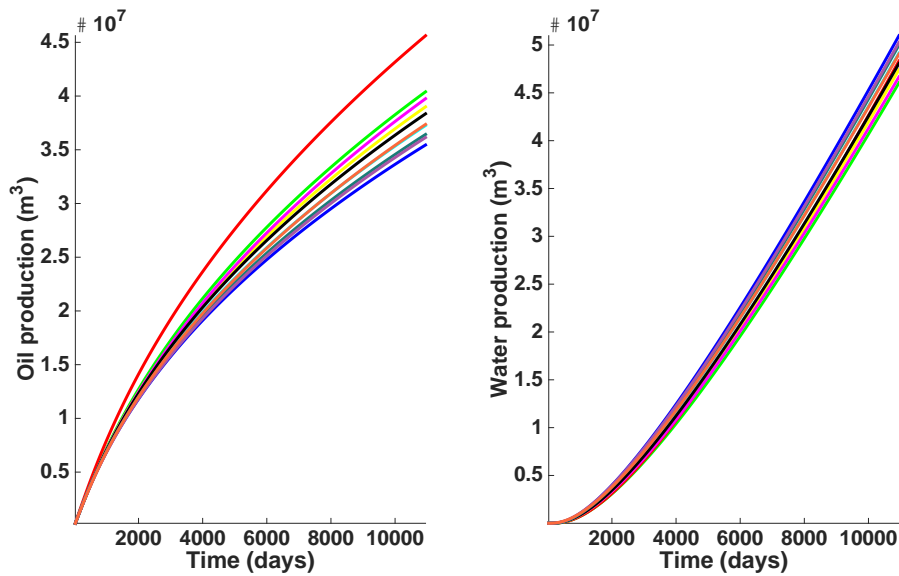


(a) Full coarsening.

(b) (x, y) -semicoarsening.(c) Structured (x, y) -semicoarsening (Cartesian semi).FIG. 12. *Modified SAIGUP agglomerates (16 fine elements per agglomerate).*

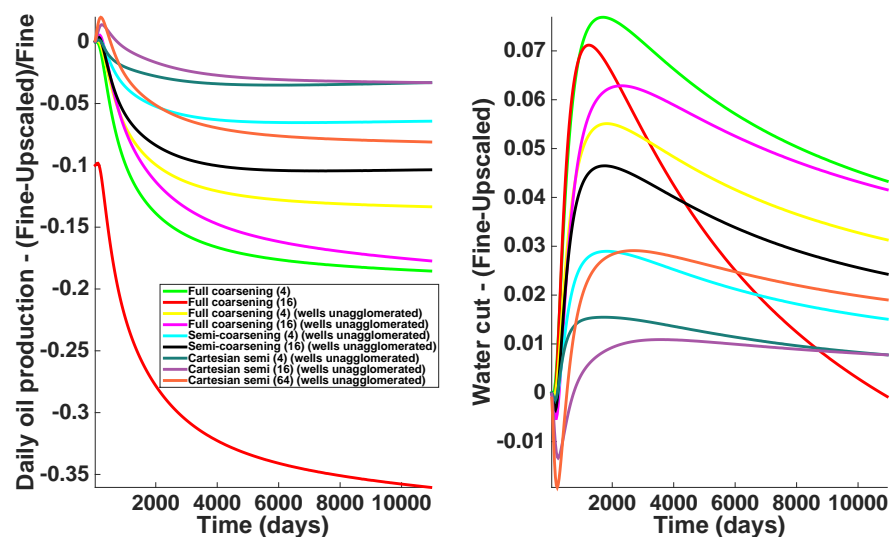


(a) Daily production data for fine-grid and upscaled models.

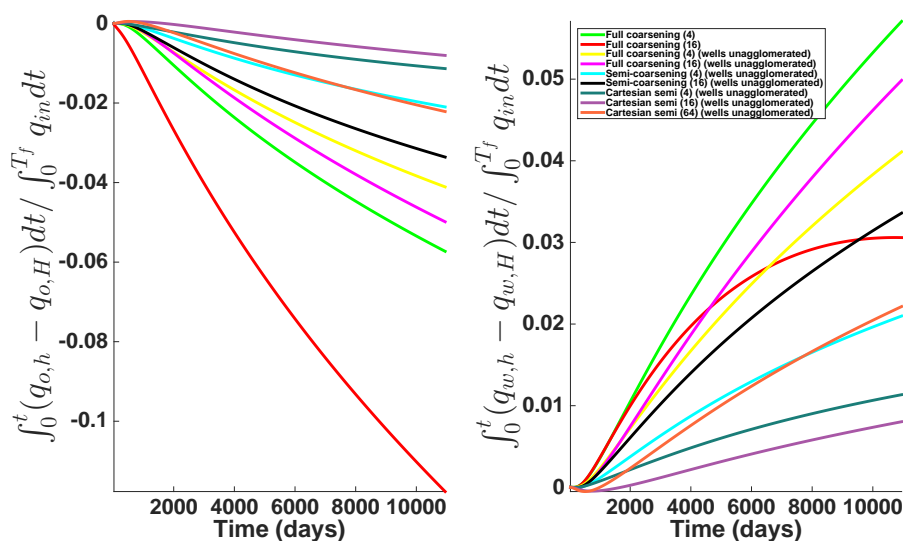


(b) Accumulated production data for fine-grid and upscaled models.

FIG. 13. Daily and accumulated production data for the modified SAIGUP model.



(a) Difference between fine-grid and upscaled models in terms of daily production and water cut.



(b) Difference in production between the fine-grid and upscaled models integrated over time relative to the total injection of water over time.

FIG. 14. Difference between fine-grid and upscaled productions for the modified SAIGUP model.

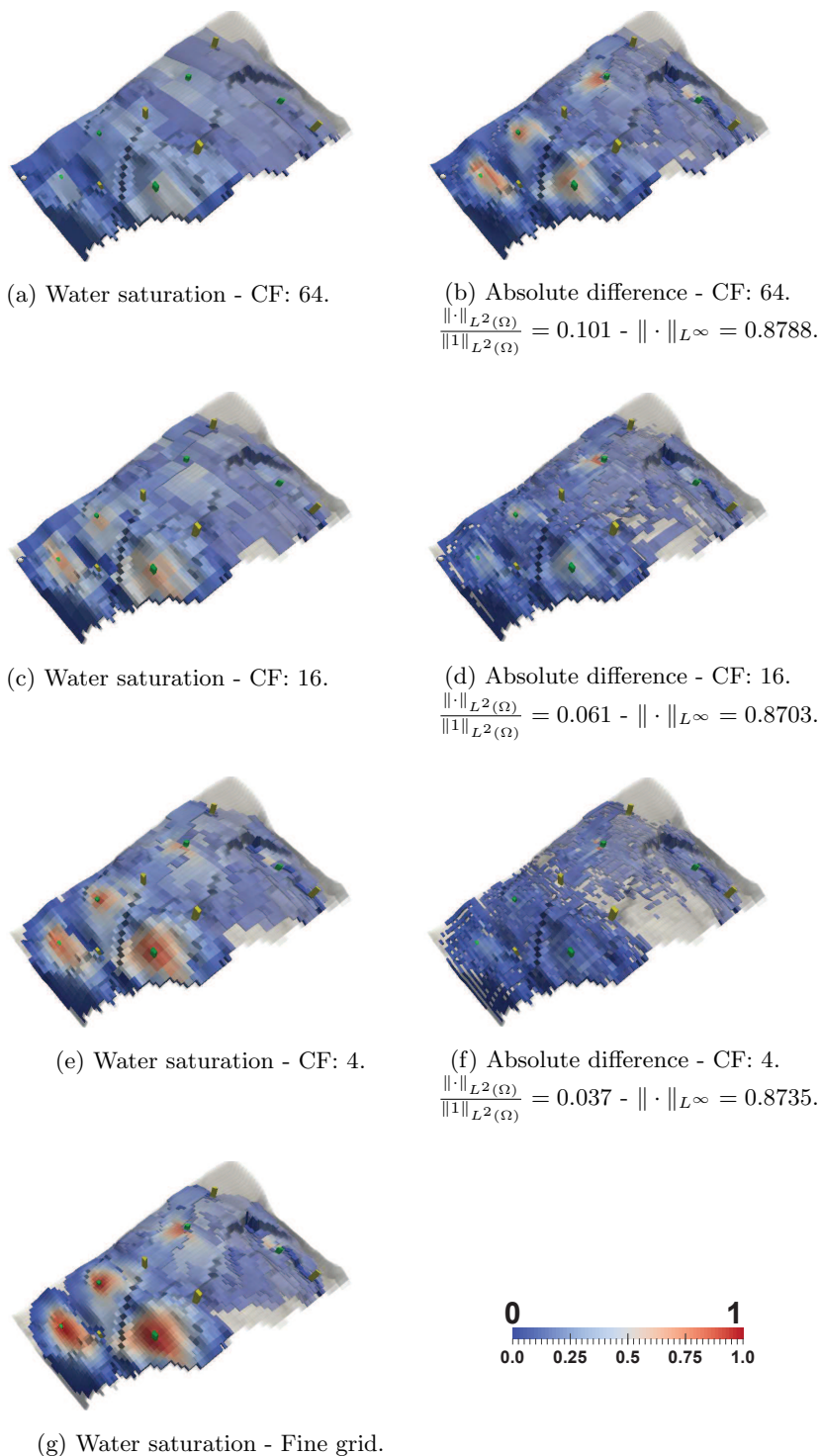


FIG. 15. Left column: Water saturation after 30 years for the fine-grid reference solution and the three upscaled solutions. Right column: Absolute difference between the fine-grid reference solution and the three upscaled solutions. CF is short for coarsening factor.

TABLE 6

Water breakthrough is defined as the time at which 1% of the produced fluids is water. The oil production error is computed based on the accumulated production per well: $\int_0^{T_f} (q_{o,h} - q_{o,H}) dt / \int_0^{T_f} q_{o_{total},h} dt$. * means elements with wells, and immediate neighbor elements of wells are left unagglomerated.

Water breakthrough (days)	Well 1	Well 2	Well 3	Well 4	Well 5
Fine grid	180	480	450	120	630
Full coarsening (4)	210	420	450	120	660
Full coarsening (16)	240	390	390	90	540
Full coarsening (4)*	210	450	480	120	690
Full coarsening (16)*	180	360	390	90	540
Semicoarsening (4)*	180	420	420	90	420
Semicoarsening (16)*	180	360	330	90	330
Cartesian semi (4)*	180	420	390	90	480
Cartesian semi (16)*	150	390	330	60	330
Cartesian semi (64)*	210	300	300	60	180
Oil production error					
Full coarsening (4)	-0.0044	-0.0385	-0.0341	-0.0041	-0.0160
Full coarsening (16)	-0.0597	-0.0458	-0.0649	-0.0160	-0.0130
Full coarsening (4)*	-0.0174	-0.0207	-0.0157	-0.0065	-0.0094
Full coarsening (16)*	-0.0018	-0.0412	-0.0254	-0.0102	-0.0059
Semicoarsening (4)*	-0.0198	-0.0048	-0.0050	-0.0039	-0.0021
Semicoarsening (16)*	-0.0388	-0.0077	-0.0071	-0.0038	0.0005
Cartesian semi (4)*	-0.0129	-0.0018	-0.0026	-0.0017	-0.0002
Cartesian semi (16)*	-0.0212	-0.0018	-0.0020	0.0044	0.0070
Cartesian semi (64)*	-0.0548	-0.0067	-0.0056	0.0119	0.0177

In Figures 14(a) and 14(b), it can be seen that the best compromise (in these tests) between complexities and accuracy is the Cartesian semicoarsening strategy with 64 fine elements per agglomerate. The maximum error in daily production is around 7% for oil and 2% for water with an operator complexity of only 1.09.

Figure 15 shows the error (compared to the fine-grid solution) in water saturation after 30 years of injection for the upscaled solutions using the Cartesian semicoarsening. As stated previously, the upscaling results in a more diffusive discretization; however, the upscaled models still capture the solution well. For this particular problem and choice of coarsening strategy, it seems the error in the infinity norm is consistently around 0.87, probably due to a problematic area around the green well in the top right corner. Table 6 provides information on the accuracy of the upscaled results for the five individual production wells. Specifically, the table reports for each well the time of water breakthrough and the total production error computed as $\int_0^{T_f} (q_{o,h} - q_{o,H}) dt / \int_0^{T_f} q_{o_{total},h} dt$, where the final time T_f is 30 years and $\int_0^{T_f} q_{o_{total},h} dt$ is the total amount of oil produced in all wells. It is evident that the tendency to predict earlier water breakthrough remains. Furthermore, the oil production errors range from 0.02% (well 5 with Cartesian semicoarsening (4)) to 6.5% (well 3 with full coarsening (16)). These errors are highly dependent on the choice of agglomerates. It is worth noting that the 6.5% error is in the case where agglomeration does not take the wells into consideration. This highlights the importance of keeping well cells (and possibly their neighbor cells) unagglomerated.

5.2.1. Multilevel recursive upscaling. In this section, the multilevel recursive upscaling capability of the AMGe method is demonstrated for the modified SAIGUP case. The premise for this study is the same as that described in section 5.1.1. METIS is used to form the agglomerated meshes. Five levels including the fine grid are used. A coarsening factor of 16 is used from level zero to level one, and a coarsening factor of four is used for the remaining agglomeration steps.

The error norms for the difference between the upscaled solutions and the fine-grid reference solution normalized by the norm of the fine-grid reference solution are plotted in Figure 16. On the x -axis of the figure we report the agglomeration factor

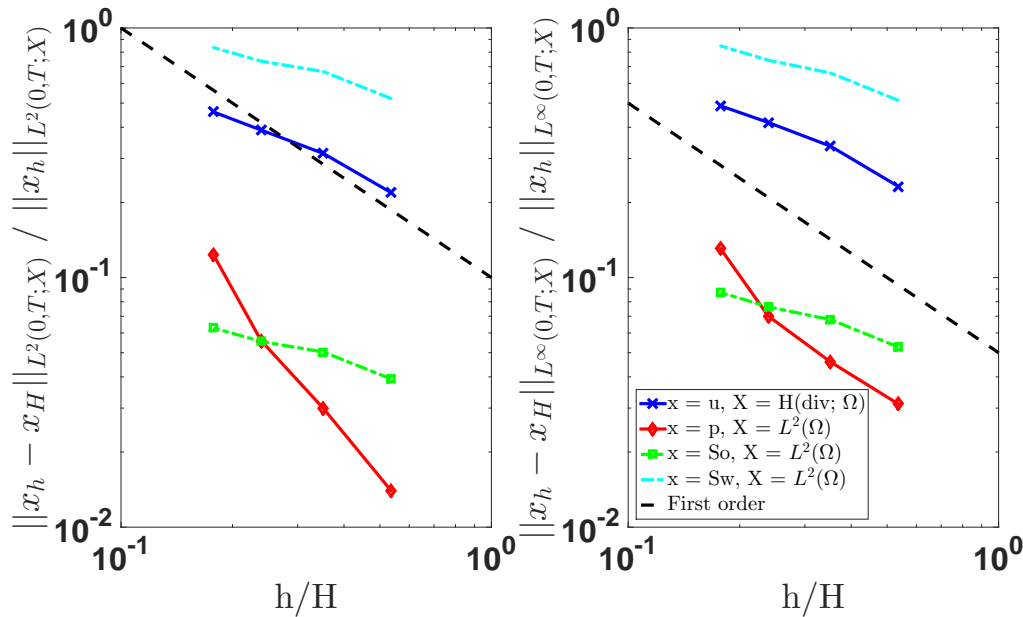


FIG. 16. Error norms (5.2) as a function of the coarsening ratio h/H , defined as the cubic root of the ratio between the number of degrees of freedom for the upscaled system and the number of degrees of freedom for the fine-grid system.

h/H roughly estimated as the cubic root of the ratio between the number of degrees of freedom of the upscaled problem and the fine-grid problem. An alternative, which is outside the scope of this paper, is to study the accuracy of the coarse solution as a function of the computational cost (time-to-solution) for the fine and upscaled models (see our results in [18]).

Figure 16 suggests that the upscaled velocity and the saturations converge linearly to the fine-grid reference solution with respect to the coarsening ratio. On the other hand, we observe larger errors for the upscaled pressure for high coarsening factors. This deterioration in the approximation properties of the pressure space may be alleviated by enriching the pressure coarse space with an additional coarse degree of freedom.

The number of degrees of freedom and complexities of the upscaled problem are given in Table 7. Considering all levels, the overall arithmetic complexity is 1.22, and the overall operator complexity is 1.80. Although the operator complexity is quite a bit larger than the SPE10 case, this result is satisfactory considering the complexity of the 3D geometry.

TABLE 7
Degrees of freedom and number of nonzeros for recursive upscaling of modified SAIGUP.

Level	#DoFs	#DoFs(u)	#DoFs(p)	nnz
0 (fine)	479736	243576	78720	3549946
1	73806	58638	5056	1864292
2	20865	16881	1328	648277
3	6550	5242	436	235884
4	2683	2023	220	102223

6. Summary. A version of the element-based algebraic multigrid (AMGe) with guaranteed approximation properties for the coarse velocity spaces has successfully been applied to upscale a mixed formulation of the incompressible reservoir simulation equations. The method has demonstrated the ability to accurately approximate the solution using significantly fewer degrees of freedom than that of the original system. More importantly, the nonzero entries of the resulting coarse (upscaled) problem, measured by the operator complexity, stay much less than two. This means that the memory requirement for storing the upscaled problem is (much) less than the storage needed for the fine-grid one. Two challenging test cases have been used to demonstrate this. Multilevel results show that the errors as a function of coarsening increase at a sublinear rate and therefore demonstrate that the AMGe coarse spaces lead to more accurate results than solving the same problem using standard finite elements on an equally coarse grid (i.e., for the same number of degrees of freedom). One of the important challenges is to accurately predict when the water reaches a production well. The upscaled simulations compute oil and water production curves that accurately approximate the ones computed on the fine grid; however, they tend to underestimate the exact time when water breakthrough happens due to the higher numerical diffusivity of the upscaled discretization. The experiments we performed have shown that the agglomeration strategy (grouping of the fine-grid elements into agglomerates to get a coarse mesh) has a large impact on the resulting upscaled approximation. It is important to leave the elements containing wells and (possibly) their immediate neighbors unagglomerated to capture the near-well flow accurately. Furthermore, due to the strong coupling in the vertical direction, only agglomerating in the x - and y -directions gives a significantly better upscaled approximation. Finally, we have demonstrated that the method can be used for multilevel upscaling to generate a hierarchy of coarser models.

7. Perspectives. In this study we have mainly focused on the accuracy of the proposed upscaling method to demonstrate the applicability of the improved AMGe method for reservoir simulation. The computational efficiency of the method is the subject of a follow-up paper, where all time-to-solution aspects, which are both hardware and implementation dependent, will be studied in depth. The computational benefits of the presented approach will be best utilized in a parallel computing setting. Improved parallelization of the software with MPI-OpenMP is an ongoing activity. These techniques are particularly well suited for modern multicore architectures, because the construction of the coarse spaces by solving many small local problems offers a high level of concurrency in the computations. Higher-order upscaling methods would further improve computational efficiency, because higher order provides more computationally intensive local operations, meaning the relative communication overhead is not as significant as it is for low order.

The solution of the coarse-grid mixed systems is also an important issue. We stress the fact that for very large-scale problems, even the coarse-grid problem may still be fairly large, and therefore direct solvers will not be a feasible approach. The coarse-grid mixed systems can in fact be more ill-conditioned than the original fine-grid system, which means solvers which work fine for the fine-grid problem (such as the block diagonal AMG preconditioner) may experience difficulties with the coarse-grid systems. This is due to the possibly complicated geometry of the agglomerates (and, possibly, the uneven distribution of coarse degrees of freedom within each agglomerate). We have ongoing activities in applying AMGe techniques to develop more robust preconditioners for the upscaled systems. In this way, the coarse spaces used

for upscaling can be reused for preconditioning purposes.

Another possible direction is to construct adaptive coarsening strategies. As we have already observed in section 4.4, our AMGe approach allows us to locally increase the spatial resolution of the upscaled solution by either locally enriching the coarse degrees of freedom in a particular agglomerated element (similar to “ p ”-refinement for finite elements) or to use smaller agglomerates (similar to “ h ”-refinement). In addition, effective a posteriori error estimators can be efficiently computed by exploiting the multilevel nature of our method.

Last but not least, future research directions include extending our framework to fully coupled (pressure-velocity-saturations) implicit-in-time integrators. This will include using our multilevel hierarchies of coarse spaces with guaranteed approximation properties needed to construct efficient nonlinear multigrid solvers (such as the full approximation scheme).

REFERENCES

- [1] J. E. AARNES, *On the use of a mixed multiscale finite element method for greater flexibility and increased speed or improved accuracy in reservoir simulation*, Multiscale Model. Simul., 2 (2004), pp. 421–439, <https://doi.org/10.1137/030600655>.
- [2] J. E. AARNES, S. KROGSTAD, AND K.-A. LIE, *A hierarchical multiscale method for two-phase flow based upon mixed finite elements and nonuniform coarse grids*, Multiscale Model. Simul., 5 (2006), pp. 337–363, <https://doi.org/10.1137/050634566>.
- [3] J. E. AARNES, V. KIPPE, AND K.-A. LIE, *Mixed multiscale finite elements and streamline methods for reservoir simulation of large geomodels*, Adv. in Water Res., 28 (2005), pp. 257–271.
- [4] J. E. AARNES, S. KROGSTAD, AND K.-A. LIE, *Multiscale mixed/mimetic methods on corner-point grids*, Comput. Geosci., 12 (2008), pp. 297–315.
- [5] T. ARBOGAST, *An overview of subgrid upscaling for elliptic problems in mixed form*, in Current Trends in Scientific Computing, Contemp. Math. 329, AMS, Providence, RI, 2003, pp. 21–32.
- [6] T. ARBOGAST, *Analysis of a two-scale, locally conservative subgrid upscaling for elliptic problems*, SIAM J. Numer. Anal., 42 (2004), pp. 576–598, <https://doi.org/10.1137/S0036142902406636>.
- [7] T. ARBOGAST AND K. J. BOYD, *Subgrid upscaling and mixed multiscale finite elements*, SIAM J. Numer. Anal., 44 (2006), pp. 1150–1171, <https://doi.org/10.1137/050631811>.
- [8] T. ARBOGAST, M. JUNTUNEN, J. POOL, AND M. F. WHEELER, *A discontinuous Galerkin method for two-phase flow in a porous medium enforcing $H(\text{div})$ velocity and continuous capillary pressure*, Comput. Geosci., 17 (2013), pp. 1055–1078, <https://doi.org/10.1007/s10596-013-9374-y>.
- [9] K. AZIZ, L. DURLOFSKY, AND H. TCHELEPI, *Notes on Petroleum Reservoir Simulation*, Department of Petroleum Engineering, School of Earth Sciences, Stanford University, Stanford, CA, 2005.
- [10] M. BENZI, G. H. GOLUB, AND J. LIESEN, *Numerical solution of saddle point problems*, Acta Numer., 14 (2005), pp. 1–137.
- [11] A. BORZI AND V. SCHULZ, *Multigrid methods for PDE optimization*, SIAM Rev., 51 (2009), pp. 361–395, <https://doi.org/10.1137/060671590>.
- [12] X. C. CAI, L. MARCINKOWSKI, AND P. S. VASSILEVSKI, *An element agglomeration nonlinear additive Schwarz preconditioned Newton method for unstructured finite element problems*, Appl. Math., 50 (2005), pp. 247–275.
- [13] T. CHARTIER, R. D. FALGOUT, V. E. HENSON, J. JONES, T. MANTEUFFEL, S. MCCORMICK, J. RUGE, AND P. S. VASSILEVSKI, *Spectral AMGe (ρ AMGe)*, SIAM J. Sci. Comput., 25 (2003), pp. 1–26, <https://doi.org/10.1137/S106482750139892X>.
- [14] T. CHARTIER, R. FALGOUT, V. E. HENSON, J. JONES, T. MANTEUFFEL, S. MCCORMICK, J. RUGE, AND P. S. VASSILEVSKI, *Spectral element agglomerate AMGe*, in Domain Decomposition Methods in Science and Engineering XVI, Lect. Notes Comput. Sci. Eng. 55, Springer, Berlin, 2007, pp. 513–521.
- [15] Z. CHEN AND T. Y. HOU, *A mixed multiscale finite element method for elliptic problems with oscillating coefficients*, Math. Comp., 72 (2003), pp. 541–576, <https://doi.org/10.1090/S0025-5718-02-01441-2>.

- [16] Z. CHEN, G. HUAN, AND B. LI, *An improved IMPES method for two-phase flow in porous media*, Transp. Porous Media, 54 (2004), pp. 361–376, <https://doi.org/10.1023/b:tipm.0000003667.86625.15>.
- [17] Z. CHEN, G. HUAN, AND Y. MA, *Computational Methods for Multiphase Flows in Porous Media*, SIAM, Philadelphia, 2006, <https://doi.org/10.1137/1.9780898718942>.
- [18] M. CHRISTENSEN, U. VILLA, AND P. VASSILEVSKI, *Multilevel techniques lead to accurate numerical upscaling and scalable robust solvers for reservoir simulation*, in SPE Reservoir Simulation Symposium, Houston, TX, 2015, pp. 1156–1167.
- [19] M. CHRISTENSEN, U. VILLA, AND P. VASSILEVSKI, *Nonlinear Multigrid Solvers Exploiting AMGe Coarse Spaces with Approximation Properties*, Tech. Report LLNL-JRNL-697795, Lawrence Livermore National Laboratory, Livermore, CA, 2016.
- [20] K. CLIFFE, M. GILES, R. SCHEICHL, AND A. L. TECKENTRUP, *Multilevel Monte Carlo methods and applications to elliptic PDEs with random coefficients*, Comput. Vis. Sci., 14 (2011), pp. 3–15.
- [21] T. J. DODWELL, C. KETELSEN, R. SCHEICHL, AND A. L. TECKENTRUP, *A hierarchical multilevel Markov chain Monte Carlo algorithm with applications to uncertainty quantification in subsurface flow*, SIAM/ASA J. Uncertainty Quantification, 3 (2015), pp. 1075–1108, <https://doi.org/10.1137/130915005>.
- [22] L. J. DURLOFSKY AND K. AZIZ, *Advanced Techniques for Reservoir Simulation and Modeling of Nonconventional Wells*, Tech. Report, 2004, <https://doi.org/10.2172/834134>.
- [23] L. J. DURLOFSKY, Y. EFENDIEV, AND V. GINTING, *An adaptive local-global multiscale finite volume element method for two-phase flow simulations.*, Adv. in Water Res., 30 (2007), pp. 576–588.
- [24] Y. EFENDIEV AND L. J. DURLOFSKY, *A generalized convection-diffusion model for subgrid transport in porous media*, Multiscale Model. Simul., 1 (2003), pp. 504–526, <https://doi.org/10.1137/S1540345902413693>.
- [25] R. D. FALGOUT, P. S. VASSILEVSKI, AND L. T. ZIKATANOV, *On two-grid convergence estimates*, Numer. Linear Algebra Appl., 12 (2005), pp. 471–494.
- [26] G. GOLUB AND C. VAN LOAN, *Matrix Computations*, The Johns Hopkins University Press, Baltimore, MD, 2012, <https://books.google.com/books?id=5U-l8U3P-VUC>.
- [27] T. Y. HOU AND X.-H. WU, *A multiscale finite element method for elliptic problems in composite materials and porous media*, J. Comput. Phys., 134 (1997), pp. 169–189.
- [28] R. FALGOUT ET AL., *HYPRE: Scalable Linear Solvers and Multigrid Methods*, Lawrence Livermore National Laboratory, Livermore, CA, 2016, <https://www.llnl.gov/casc/hypre/>.
- [29] P. JENNY, S. LEE, AND H. TCHELEPI, *Multi-scale finite-volume method for elliptic problems in subsurface flow simulation*, J. Comput. Phys., 187 (2003), pp. 47–67.
- [30] J. E. JONES AND P. S. VASSILEVSKI, *AMGe based on element agglomeration*, SIAM J. Sci. Comput., 23 (2001), pp. 109–133, <https://doi.org/10.1137/S1064827599361047>.
- [31] D. KALCHEV, C. KETELSEN, AND P. S. VASSILEVSKI, *Two-level adaptive algebraic multigrid for a sequence of problems with slowly varying random coefficients*, SIAM J. Sci. Comput., 35 (2013), pp. B1215–B1234, <https://doi.org/10.1137/120895366>.
- [32] G. KARYPIS AND V. KUMAR, *A fast and high quality multilevel scheme for partitioning irregular graphs*, SIAM J. Sci. Comput., 20 (1998), pp. 359–392, <https://doi.org/10.1137/S1064827595287997>.
- [33] S. KNAPEK, *Matrix-dependent multigrid homogenization for diffusion problems*, SIAM J. Sci. Comput., 20 (1998), pp. 515–533, <https://doi.org/10.1137/S1064827596304848>.
- [34] S. KROGSTAD, K. LIE, AND B. SKAFLESTAD, *Mixed multiscale methods for compressible flow*, in Proceedings of the 13th European Conference on the Mathematics of Oil Recovery (ECMOR XIII), Biarritz, France, 2012.
- [35] M. S. LAMINE, S. KROGSTAD, K.-A. LIE, M. PAL, ET AL., *Multiscale method for simulating two-and three-phase flow in porous media*, in Proceedings of the SPE Reservoir Simulation Symposium, Woodlands, TX, 2013.
- [36] I. LASHUK AND P. VASSILEVSKI, *Element agglomeration coarse Raviart–Thomas spaces with improved approximation properties*, Numer. Linear Algebra Appl., 19 (2012), pp. 414–426.
- [37] I. LASHUK AND P. S. VASSILEVSKI, *On some versions of the element agglomeration AMGe method*, Numer. Linear Algebra Appl., 15 (2008), pp. 595–620.
- [38] I. LASHUK AND P. S. VASSILEVSKI, *The construction of the coarse de Rham complexes with improved approximation properties*, Comput. Methods Appl. Math., 14 (2014), pp. 257–303.
- [39] K. LIPNIKOV, J. D. MOULTON, AND D. SVYATSKIY, *A multilevel multiscale mimetic (M^3) method for two-phase flows in porous media*, J. Comput. Phys., 227 (2008), pp. 6727–6753.
- [40] K. LIPNIKOV, J. D. MOULTON, AND D. SVYATSKIY, *A multiscale multilevel mimetic (M^3) method for well-driven flows in porous media*, Procedia Comput. Sci., 1 (2010), pp. 771–779, <https://doi.org/10.1016/j.procs.2010.04.083>.

- [41] K. LIPNIKOV, J. D. MOULTON, AND D. SVYATSKIY, *Adaptive strategies in the multilevel multiscale mimetic (M^3) method for two-phase flows in porous media*, Multiscale Model. Simul., 9 (2011), pp. 991–1016, <https://doi.org/10.1137/100787544>.
- [42] S. P. MACLACHLAN AND J. D. MOULTON, *Multilevel upscaling through variational coarsening*, Water Resour. Res., 42 (2006), W02418, <https://doi.org/10.1029/2005WR003940>.
- [43] K.-A. MARDAL AND R. WINTHER, *Preconditioning discretizations of systems of partial differential equations*, Numer. Linear Algebra Appl., 18 (2011), pp. 1–40, <https://doi.org/10.1002/nla.716>.
- [44] MFEM, *Finite Element Discretization Library*, 2017, mfem.org.
- [45] J. MIKYŠKA AND A. FIROOZABADI, *Implementation of higher-order methods for robust and efficient compositional simulation*, J. Comput. Phys., 229 (2010), pp. 2898–2913, <https://doi.org/10.1016/j.jcp.2009.12.022>.
- [46] J. MOORTGAT, S. SUN, AND A. FIROOZABADI, *Compositional modeling of three-phase flow with gravity using higher-order finite element methods*, Water Resour. Res., 47 (2011), W05511.
- [47] O. MØYNER AND K.-A. LIE, *A multiscale two-point flux-approximation method*, J. Comput. Phys., 275 (2014), pp. 273–293.
- [48] M. PAL, S. LAMINE, K.-A. LIE, AND S. KROGSTAD, *Multiscale method for two and three-phase flow simulation in subsurface petroleum reservoirs*, in Proceedings of the 13th European Conference on the Mathematics of Oil Recovery (ECMOR XIII), Biarritz, France, 2012.
- [49] J. E. PASCIAK AND P. S. VASSILEVSKI, *Exact de Rham sequences of spaces defined on macroelements in two and three spatial dimensions*, SIAM J. Sci. Comput., 30 (2008), pp. 2427–2446, <https://doi.org/10.1137/070698178>.
- [50] D. PEACEMAN, *Interpretation of well-block pressures in numerical reservoir simulation with nonsquare grid blocks and anisotropic permeability*, Soc. Pet. Eng. J., 23 (1983), pp. 531–543.
- [51] SOCIETY OF PETROLEUM ENGINEERS, *Tenth SPE Comparative Solution Project*, 2001, <http://www.spe.org/web/csp/>.
- [52] UCD FAULT ANALYSIS GROUP, *Saigup—Sensitivity Analysis of the Impact of Geological Uncertainties on Production Forecasting in Clastic Hydrocarbon Reservoirs*, 2000–2004, <http://www.fault-analysis-group.ucd.ie/Projects/SAIGUP.html>.
- [53] P. S. VASSILEVSKI AND U. VILLA, *A mixed formulation for the Brinkman problem*, SIAM J. Numer. Anal., 52 (2014), pp. 258–281, <https://doi.org/10.1137/120884109>.
- [54] P. S. VASSILEVSKI, *Sparse matrix element topology with application to AMG(e) and preconditioning*, Numer. Linear Algebra Appl., 9 (2002), pp. 429–444.
- [55] P. S. VASSILEVSKI, *Multilevel Block-Factorization Preconditioners: Matrix-Based Analysis and Algorithms for Solving Finite Element Equations*, Springer, New York, 2008.
- [56] P. S. VASSILEVSKI, *Coarse spaces by algebraic multigrid: Multigrid convergence and upscaling error estimates*, Adv. Adapt. Data Anal., 3 (2011), pp. 229–249, <https://doi.org/10.1142/S1793536911000830>.
- [57] P. S. VASSILEVSKI AND U. VILLA, *A block-diagonal algebraic multigrid preconditioner for the Brinkman problem*, SIAM J. Sci. Comput., 35 (2013), pp. S3–S17, <https://doi.org/10.1137/120882846>.
- [58] T. WALLSTROM, S. HOU, M. A. CHRISTIE, L. J. DURLOFSKY, D. H. SHARP, AND Q. ZOU, *Application of effective flux boundary conditions to two-phase upscaling in porous media*, Transp. Porous Media, 46 (2002), pp. 155–178, <https://doi.org/10.1023/A%3A1015003527103>.
- [59] H. ZHOU, S. H. LEE, AND H. A. TCHELEPI, *Multiscale finite-volume formulation for saturation equations*, Soc. Pet. Eng. J., 17 (2012), SPE-119183-PA, <https://doi.org/10.2118/119183-PA>.